Star Signature Series

NB 24-10/15

USER'S MANUAL

Federal Communications Commission
Radio Frequency Interference
Statement

This equipment generates and uses radio frequency energy and if not installed and used properly, that is, in strict accordance with the manufacturer's instructions, may cause interference to radio and television reception. It has been type tested and found to comply with the limits for a Class B computing device in accordance with the specifications in Subpart J of Part 15 of FCC Rules, which are designed to provide reasonable protection against such interference in a residential installation. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- •Reorient the receiving antenna
- •Relocate the computer with respect to the receiver
- •Move the computer away from the receiver
- •Plug the computer into a different outlet so that computer and receiver are on different branch circuits.

If necessary, the user should consult the dealer or an experienced radio/television technician for additional suggestions. The user may find the following booklet, prepared by the Federal Communications Commission helpful: "How to Identify and Resolve Radio-TV Interference Problems." This booklet is available from the U.S. Government printing Office, Washington, D.C., 20402, Stock No. 004-000-00345-4.

For compliance with Federal Noise Interference Standard, this equipment requires a shielded cable.

This statement will be applied only for the printers marketed in U.S.A.

Self Declaration

Radio interferences regarding this equipment has been eliminated according to Vfg 1046/1984 announced by the DBP.
DBP has been informed about the introduction of this special equipment and has been conceded the right to examine the whole series.
It is upon the responsibility of the user to assume that his own assembled system is in accordance with the technical regulations under Vfg 1046/1984.
To observe FTZ-regulations it is necessary, to establish all connections to the printer with shielded cable.
The equipment may only be opened by qualified service representatives.

This statement will be applied only for the printers marketed in West Germany.

Trademark Acknowledgement

NB24-10, NB24-15: Star Micronics Co., Ltd.
Apple Ⅱ: Apple computer Inc.
EasyWriter: Information Unlimited Software, Inc.
IBM PC, IBM Graphics printer, IBM Proprinter: International Business Machines Corp.
Lotus 1-2-3: Lotus Development Corp.
LQ 1000: Seiko Epson Corp.
Microsoft BASIC: Microsoft Corporation
WordStar: MicroPro International Corporation

NOTICE
- •All rights reserved. Reproduction of any part of this manual in any form whatsoever, without STAR's express permission is forbidden.
- •The contents of this manual are subject to change without notice.
- •All efforts have been made to ensure the accuracy of the contents of this manual at the time of going to press. However, should any errors be detected, STAR would be greatly appreciate being informed of them.
- •The above notwithstanding, STAR can assume no responsibility for any errors in this manual.

A Special Message to the New Owner

Congratulations for choosing this printer, one of the finest available at any price. Your new printer will reward you with many years of high-quality document printing, and is backed by Star Micronics' more than 30 years in manufacturing precision parts. You already know that it represents the state of the art in tack-sharp dot matrix printing with its rugged 24-pin print head. Its bidirectional, logic-seeking operation gives you Letter-Quality hardcopy at 72 characters per second, and draft-quality print at speeds of up to 216 characters per second.

Its many features include:

- High-resolution graphics
- Built-in expandable buffer
- An entire range of character sets for printing in thirteen languages
- Choice of various typestyles
- Easy, front panel operation
- Automatic paper feed

And, best of all, your new printers is compatible with the most popular personal computers and software packages — so no matter how sophisticated your applications become, it will continue to meet your needs.

These are just a few of the reasons you chose a product by Star Micronics, one of the most respected names in dot matrix printers. We want you to get the most from your printer with the least effort on your part, so please take a few minutes to read this easy-to-understand guide. You'll find that it will help you produce crisp, clean documents in the shortest time possible.

You will also find that this printer is remarkably simple to set up and operate, and that it's virtually maintenance-free. Enjoy your new Star Micronics printer!

Table of Contents

CHAPTER 1

SETTING UP YOUR PRINTER

Subjects covered in Chapter 1 include—
- **Choosing a suitable place for your printer**
- **Unpacking**
- **Setting up**

LOCATING THE PRINTER

Give some thought to the best place to put the printer. Both the printer and computer should be used in normal office surroundings. For best performance, we recommend:

- Use the printer on a flat surface.
- Keep it out of direct sunlight and away from heat-producing appliances.
- Use it only in temperatures where you are comfortable.
- Avoid locations with dust, grease, or high humidity.
- Supply it "clean" electricity. Don't connect it to the same circuit used by large, noise-producing appliances (such as refrigerators).
- Make sure the line voltage is within 10% of the voltage specified on the identification plate.

CHECK THE CARTON CONTENTS

Open the carton and check each item in the box against Figure 1-1 to make sure that you have everything (there should be six items).
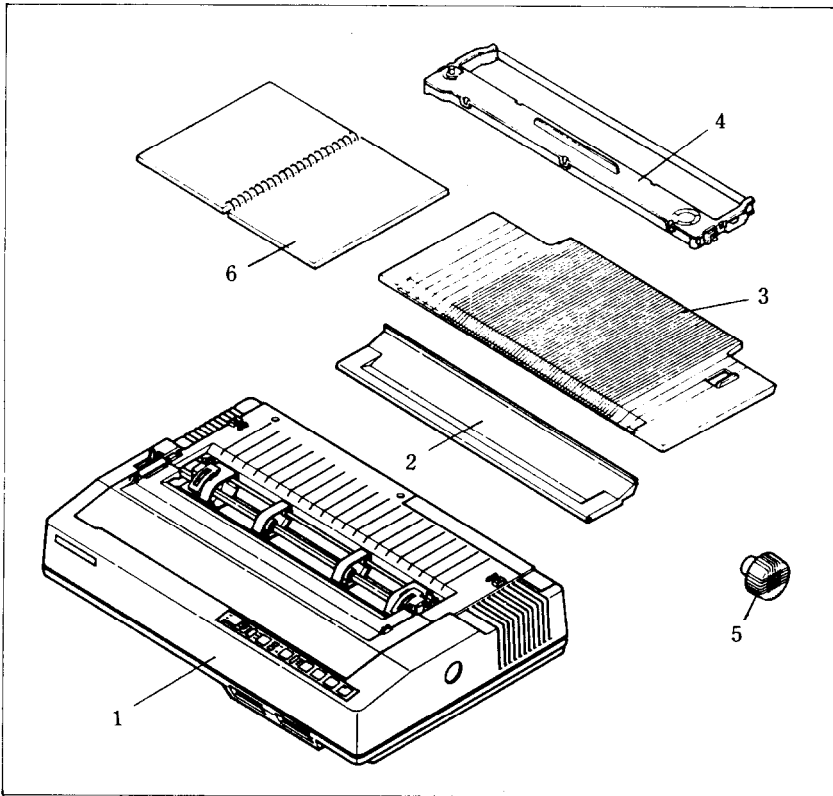
Figure 1-1. Check to make sure you have all six items: 1) Printer, 2) Mute cover, 3) Paper guide, 4) Ribbon cartridge, 5) Platen knob, and 6) User's manual.

You should also have a parallel interface board to connect your computer to the printer. Also available are a parallel interface board with extra buffer memory and a serial interface board, both optional. (More on interfaces later.)

■ Removing the printer cover

The printer's cover is important for two reasons — it keeps dust and dirt away from the printer's delicate mechanism, and it absorbs nearly all of the printer's operating sounds. Don't take off the cover unless you have to change the ribbon or make an adjustment.

To remove the cover, lift up the back of the cover to

disengage the two or three tabs at the front, then lift it off the rest of the way. To replace the cover, slide the tabs in at the front and lower it into place. Figure 1-2 shows the proper position and movement for both removing and replacing the cover.
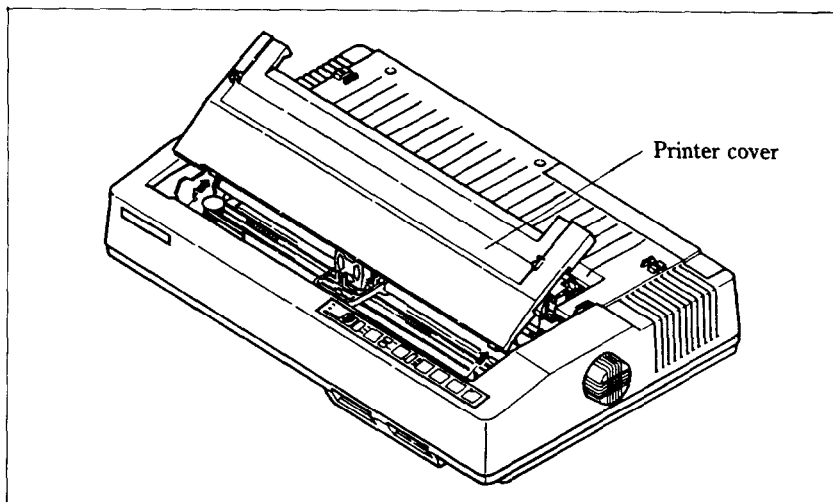
Figure 1-2. Remove the printer cover by lifting carefully.

■ Removing the packing tube

The printer is shipped with a protective spiral tube to keep the print head from being damaged during shipping. This tube
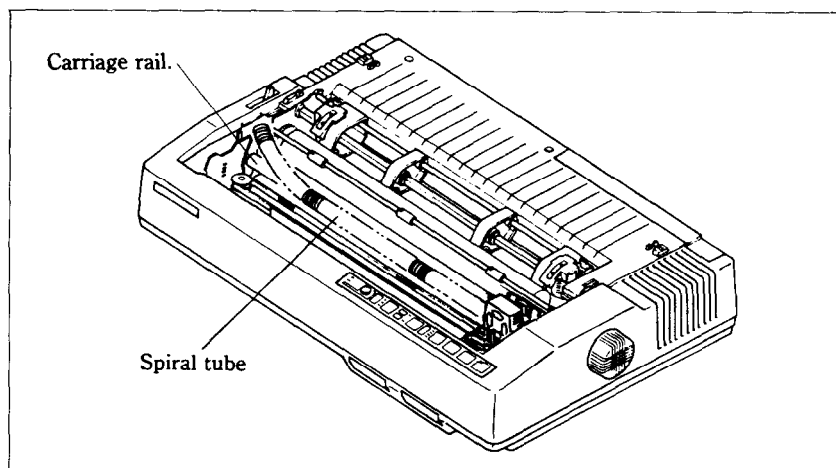
Figure 1-3. Remove the protective tube from the carriage rail.

must be removed before use. First, take off the printer cover. The protective tube is split along its bottom side to allow easy removal. Grasp one end of the spiral tube and firmly pull it upward and away from the carriage rail.

■ Removing shipping screws (15-inch type only)

There are two shipping screws inside the 15-inch type printer, used to hold the internal chassis securely to the lower case during shipping. To get at these, remove the printer cover from the printer. Then remove the red screws with a Phillips screwdriver as shown in Figure 1-4.

You'll be smart to save these screws, along with the rest of the packing material and the shipping carton, in case you ever have to ship the printer. Tape the screws somewhere on the carton or packing.
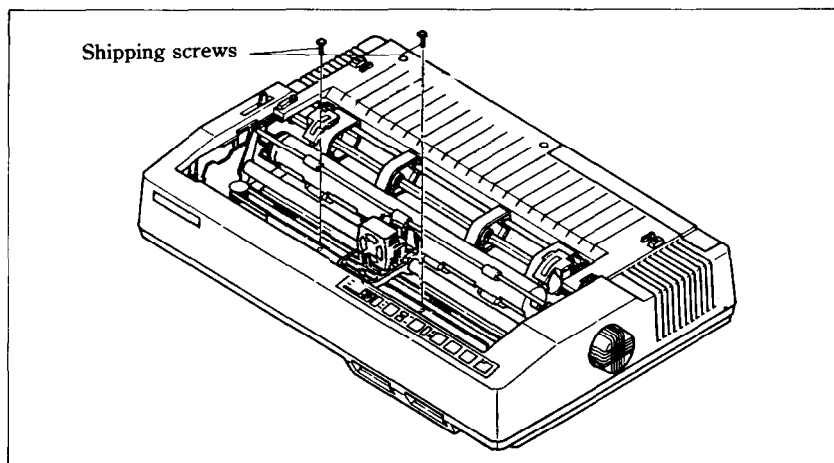
Figure 1-4. There are two shipping screws inside the printer, during shipping. They should be removed before use.

■ Fitting the platen knob (15-inch type only)

The platen knob fits into the hole in the right side of the printer case. Look at the hole in the knob. Match the flat surface in the knob with the flat side of the shaft, and press the knob onto the shaft firmly. Turn the knob a few times to make sure that the platen rotates smoothly and easily.

Caution: The platen knob is packed into a recess of the white

foam packing material which held your printer inside the pack-ing box. Be careful to remove the knob before disposing of the packing pieces.

■ Installing the ribbon cartridge

   This printer uses a neat, easy-to-change ribbon cartridge with automatic threading. To fit or change the ribbon cartridge:
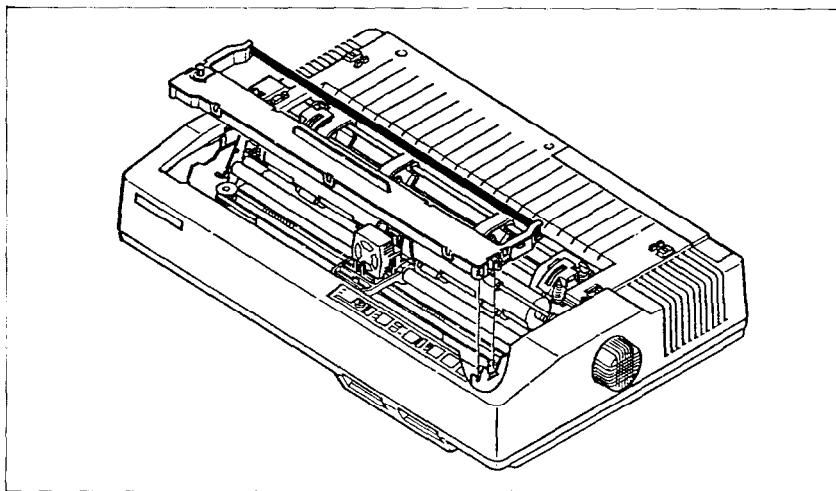
Figure 1-5.   Press the cartridge into place until the holding springs snap into place.

1. Turn *off* the power and remove the printer cover.
2. Slide the print head gently to the center of the printer. **Warning:** The print head gets hot during operation, so let it cool off before you touch it.
3. Set the release lever to either single sheet (■) or sprocket-feed paper (■).
4. With the ribbon facing away from you, use the guide holders as a fulcrum, and lightly press the cartridge down until the two holder springs snap shut to hold the cartridge firmly in place. See Figure 1-5.
5. Check that the cartridge fits so that the drive pins engage the cartridge teeth, by turning the ribbon ad-vance knob. Advance the ribbon to take up any slack.
6. Gently slide the print head carriage manually all the way

to your right or left until the ribbon automatically slips down into its proper place between the print head and the silver ribbon guide.
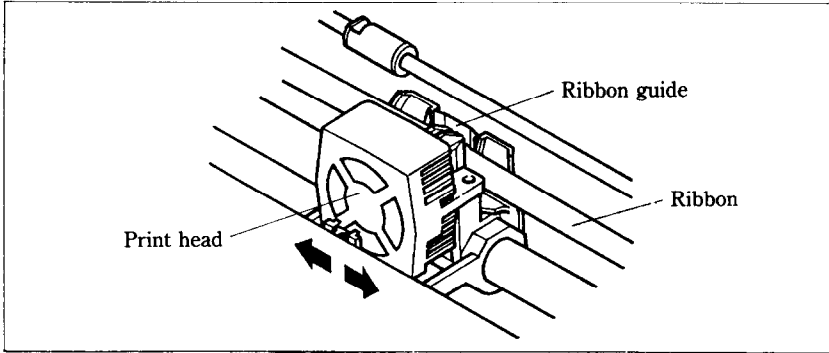
Figure 1-6. To thread the ribbon, slide the print head carriage manually to the right or left, and the ribbon will slip down by itself into its proper position.

7. Replace the printer cover. (A switch on the printer prevents the printer from operating when the cover is off.)

■ Installing the mute cover
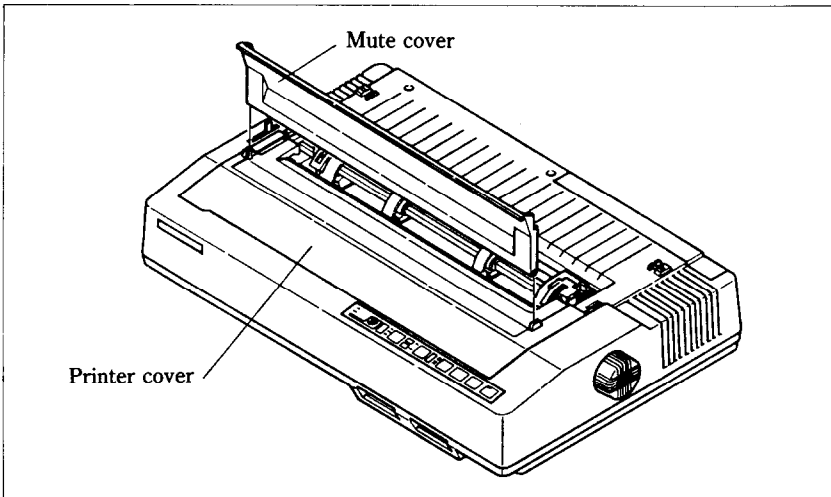
Figure 1-7. Install the mute cover by holding vertically, then lay down on the printer cover.

The mute cover reduces the printer's operating sounds.

Note the position of the guide pins on the mute cover as shown in Figure 1-7. Holding the mute cover at each end, insert the guide pins into the cutout hooks on the printer cover. Hold the mute cover down onto the printer.

To remove it, reverse the procedure.

■ Installing the interface board
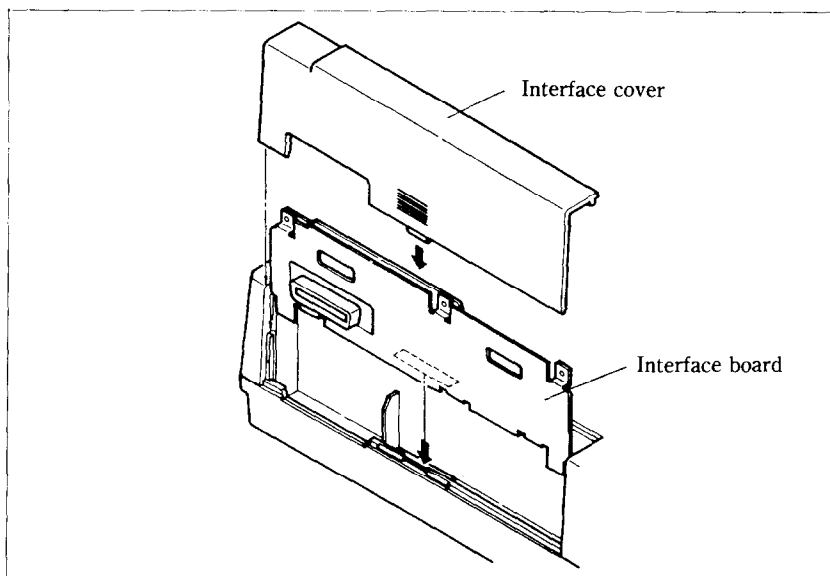
Figure 1-8. Slide the I/F board into the slot.

A parallel interface board comes standard with your printer. In addition, there are two optional interface (I/F) boards: a serial interface board that lets you connect to a serial (RS-232C) computer port; and the other board contains extra buffer memory for parallel output. All three boards are installed the same way.

To install or change an I/F board, turn off the power and turn the printer around. Notice the snap-on cover at the left end. Take it off by sliding it upwards.

Fit the I/F board into the side slots and slide it all the way down (don't force it). Make sure that the board connector is seated snugly in its socket. Put the cover back on and connect the cable from your computer — you're ready to go.

■ Installing the font cartridge

This printer is mounted the Prestige characters as standard.

You can add one or two different character styles by installing the optional font cartridges to the printer.

To install or change a font cartridge, start by turing off the power switch.

Fit the font cartridge into the slot and slide it all the way.
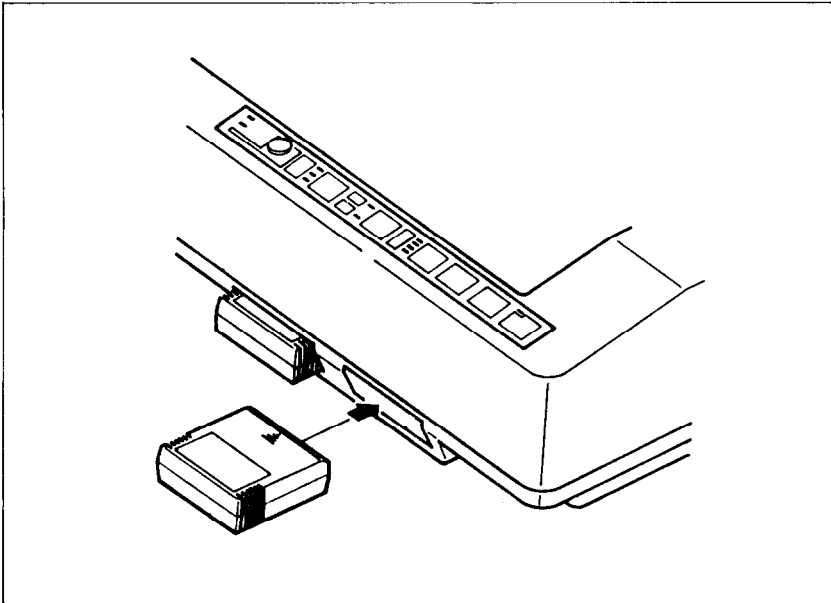
Figure 1-9. Slide the font cartridge into front slot while the power switch off.

CHAPTER 2
GETTING TO KNOW YOUR PRINTER

Subjects covered in Chapter 2 include—
- **Parts of the printer — what they're for and how to use them**
- **Paper selection and loading**
- **Adjustment**
- **Testing printer operation**

CONTROLS AND PARTS OF THE PRINTER

■ Parts of the printer

PRINTER COVER — This protects the ribbon and the print head from dust and dirt, and cuts down the sound of the printer.

MUTE COVER — This further reduces the sound of the printer.

PAPER GUIDE — This flat plastic molding guide has two functions, depending on what kind of paper you are using. If you are using single sheets, the guide is propped up on top of the printer, and serves as a guide. If you are using sprocket-feed paper, the guide is reversed and laid flat on top of the printer, and serves as a paper separator.

POWER CORD — This cord connects the printer to its power source, usually a wall outlet. It's located at the left rear of the printer.

PRINT HEAD — This is the device that does the actual printing. Like the strike lever in a typewriter, tiny pins in the print head hit the paper through a ribbon.

TRACTOR FEED UNIT — The drive gear and sprockets of the tractor feed unit move sprocket-feed paper through the
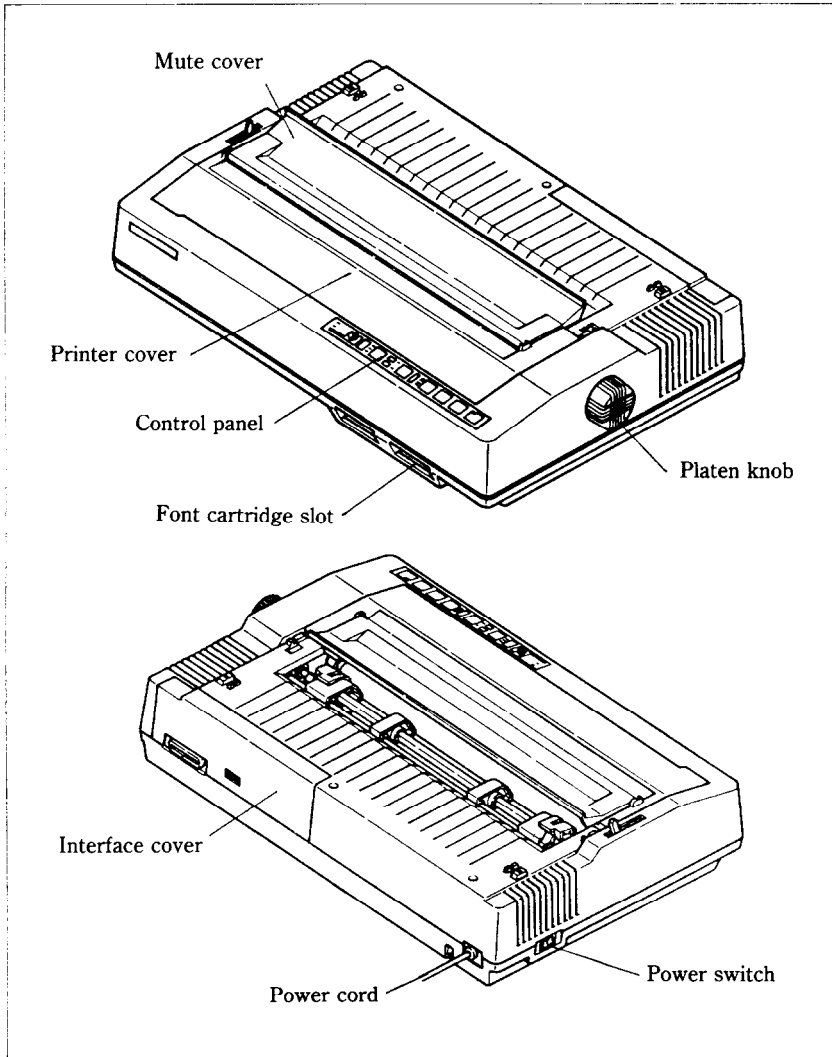
Figure 2-1. Front and rear views of the printer.

printer.

PLATEN — This is the rubber cylinder that carries paper to the print head.

INTERFACE COVER — On the back of the printer, this cover conceals and protects the interface board. The cover is cut away to expose the external connector to which you attach

your computer cable.

SLOTS FOR FONT CARTRIDGES — These are the two openings at the bottom right of the front of the printer. Inserting an optional font cartridge pushes the small door up out of the way and exposes a connector which accepts the cartridge.

■ Controls and indicators

Now let's go over the controls and indicators of the printer, starting with the control panel on the right. There are twelve indicators and seven keys on the control panel.

[15 inch type]

[10 inch type]
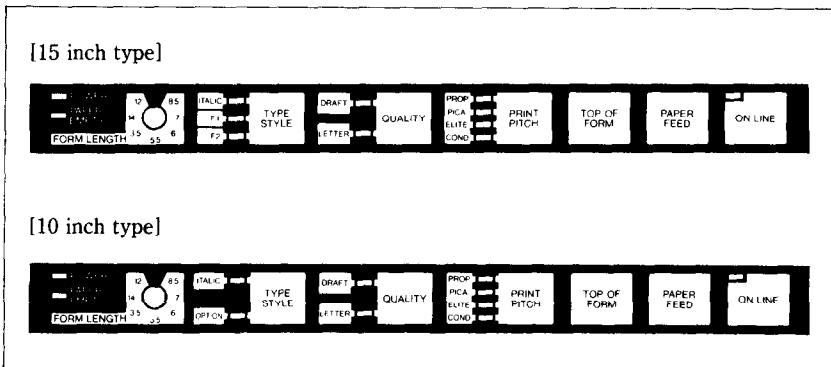
Figure 2-2. The control panel.

POWER INDICATOR — Glows green when the power is *on*.
PAPER EMPTY INDICATOR — Glows red when the printer runs out of paper and when certain printer malfunctions occur.
FORM LENGTH SWITCH — Tells the printer how long the pages are.
TYPE STYLE INDICATORS — Glows green to indicate the character set in use as shown below (set by the Type Style key or the DIP switch).

Indicator to glow	Type Style
None	Roman Internal Characters
Italic	Italic Internal Characters
F1	Roman Font 1 Characters
F1 and Italic	Italic Font 1 Characters
F2	Roman Font 2 Characters (15″ type only)
F2 and Italic	Italic Font 2 Characters (15″ type only)

TYPE STYLE KEY — Changes the type style among four or six styles when the On Line indicator is off.

QUALITY INDICATORS — Either DRAFT or LETTER glows green to indicate the character quality (set by the Quality key).

QUALITY KEY — Changes the print quality every time it is pressed when the On Line indicator is off.

PITCH INDICATORS — Shows the character pitch currently set as shown below.

Indicator to glow	Print Pitch
Pica	Pica pitch (10 CPI)
Elite	Elite pitch (12 CPI)
Cond	Semi-condensed pitch (15 CPI)
Pica and Cond	Pica condensed pitch (17 CPI)
Elite and Cond	Elite condensed pitch (20 CPI)
Prop and Pica	Pica proportional pitch
Prop and Elite	Elite proportional pitch
Prop and Cond	Semi-condensed proportional pitch (with control code only)
Prop, Pica and Cond	Pica condensed proportional pitch (with control code only)
Prop, Elite and Cond	Elite condensed proportional pitch (with control code only)

PRINT PITCH KEY — Selects one of the seven available character pitches when the On Line indicator is off as shown above.

TOP OF FORM KEY — Ejects the current sheet, feeds a new sheet, and moves the print head to the left margin when the On Line indicator is off.

PAPER FEED KEY — Advances the paper one line at a time when the On Line indicator is off. If you hold the key down, you get consecutive line feeds.

ON LINE KEY and INDICATOR — Glows green when the printer is ready to receive data from your computer (on line). When the printer is off line, it sends a signal to the computer indicating that it cannot accept data. Pressing the On Line key switches the printer between on line and off line.

■ Extra functions

There are twelve more functions that are not directly specified on the control panel. You can access six of them by pressing one particular key when you turn on the power; six others are activated by pressing two control keys at once.
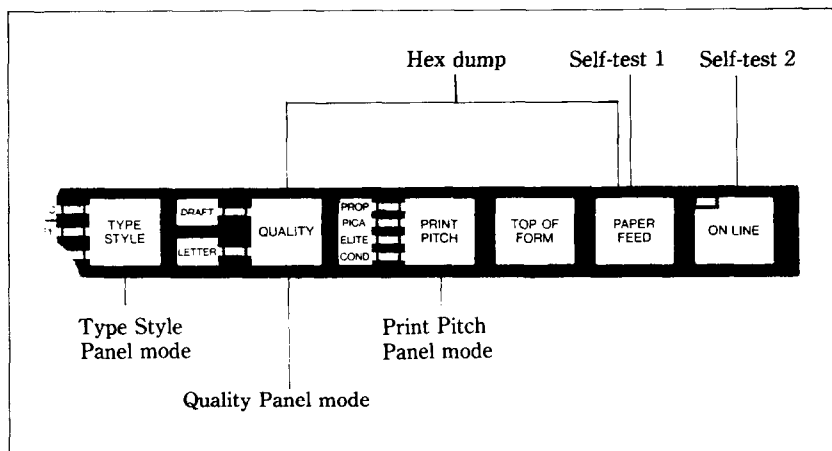
Figure 2-3. Extra functions while turning on the printer.
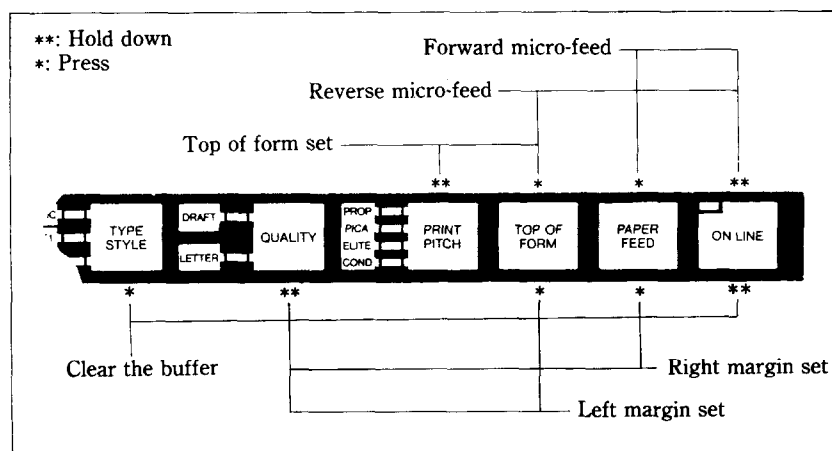
Figure 2-4. You can set many functions by the combinations of the control panel keys while in the Off Line mode.

For details, please refer to the end of next chapter.

■ Other controls

There are other controls, not connected to the control panel. Some of the more important ones are:

POWER SWITCH — The switch that turns the printer on and off is at the back, on the left side.

PLATEN KNOB — This knob is at the middle of the right side and lets you turn the platen by hand, like a typewriter.

Caution: Turn the platen knob only with the Power switch *off*. Turning the knob with the power on could damage the drive gears.

RELEASE LEVER — The release lever is on top of the printer, near the left rear corner. It controls how strongly the paper is held against the platen. The release lever has four positions: the top setting is used for inserting paper, the next is for single sheets, the third is for sprocket-feed paper, and the bottom one is used when adjusting the paper.

PAPER BAIL — The bail is the movable bar that holds the paper against the platen. Its position depends on the setting of the release lever.

DIP SWITCHES — There are one set of ten switches and one set of eight switches that are used in interfacing the printer to your particular computer. There are also switches to set print pitch, line spacing, and perforation skipping. See Appendix A for a complete list and explanation.

SELECTING AND LOADING PAPER

Your printer handles two basic types of paper — single sheets (stationery, envelopes, multipart business forms, etc.) and continuous paper (fan-fold perforated paper).

The release lever controls the pressure of the paper against the platen. It has four settings:

- ▇ — The top setting is used when you want to load paper in the printer.
- ▙ — The second setting is for normal single sheets.
- ▙ — The third is for sprocket-feed paper.
- ▥ — The bottom one is used when you want to release the paper completely to adjust it.

The paper bail holds the paper against the platen according to the setting of the release lever. The bail is opened when the release lever is in the top position, closed when the lever is in the second or third positions, and opened by the printer when the lever is in the bottom position.
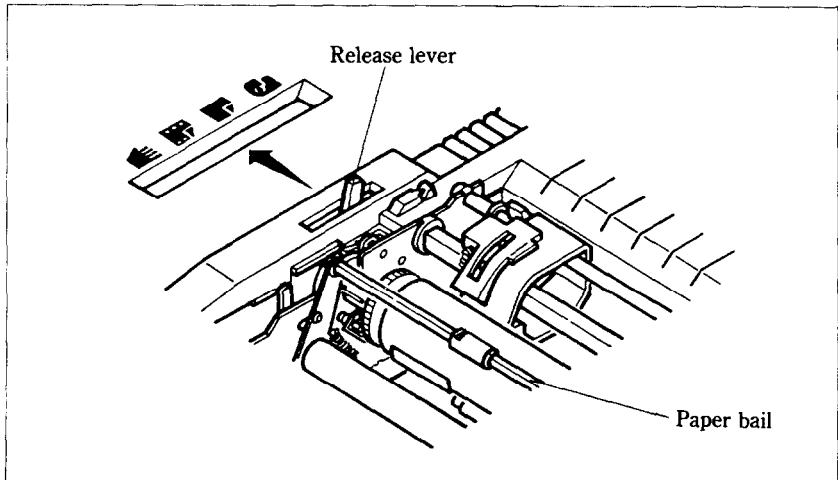
Figure 2-5. Use friction feed for single sheets and tractor feed for continuous paper.

■ Loading single sheets

Instead of feeding the paper in *manually* by turning the platen knob, you can use the release lever for automatic feeding (the printer must be turned *on*). Single sheets from 5.5 to 8.5 inches wide (6 to 14.5 inches for the 15-inch type) can be used and the paper guide must be fitted:

1. Flip the clamp levers backward and move the sprockets and sheet guides on the tractor feed unit all the way to the ends.
2. With the ribs of the paper guide toward you and the sliding adjuster at the bottom left, insert the guide into the slot at the back of the printer cover.
3. Insert the guide stand in the notches provided for it and prop up the paper guide.
4. Position the adjuster at the triangle marked on the guide for standard margins, or set it according to your own requirements.
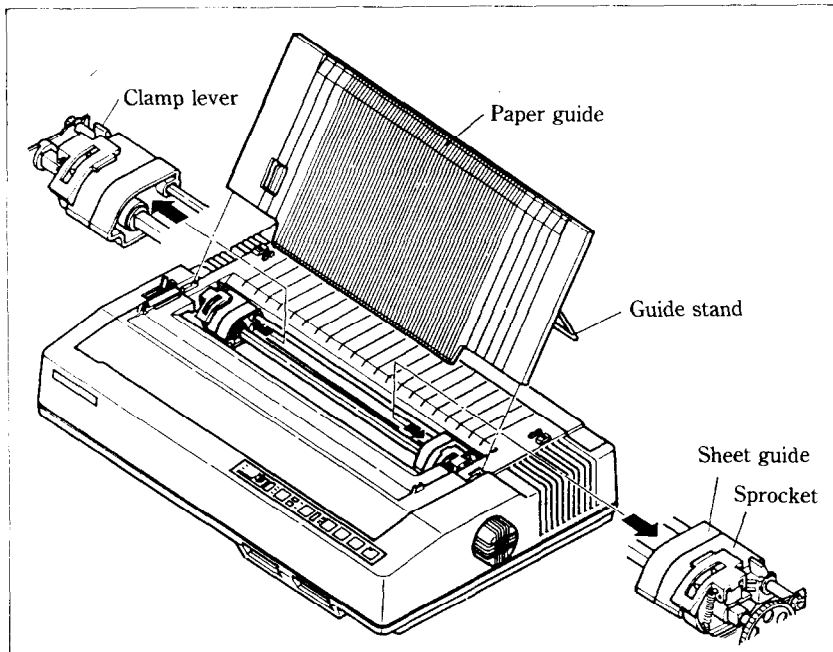
Figure 2-6. Raise the paper guide for loading single sheets.
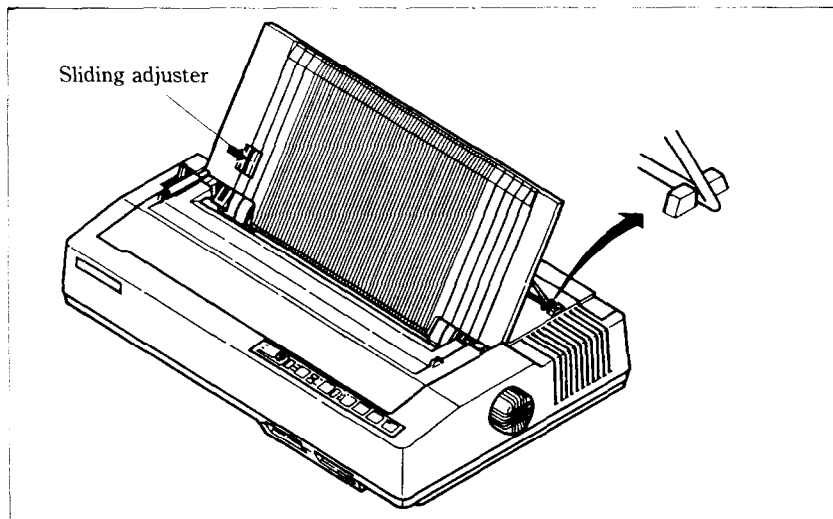
Figure 2-7. Prop up the paper guide by inserting the guide stand in the notches provided for it.

5. Set the release lever for single sheets — the second position from the top.
6. Align a sheet of paper on the guide along the adjuster.
7. Turn on the power switch. (The Paper Empty indicator will glow because there is no paper.)
8. Now set the release lever to the auto-feed setting — the top position. The printer will advance the paper automatically.
9. When the paper stops, set the release lever for single sheets again.
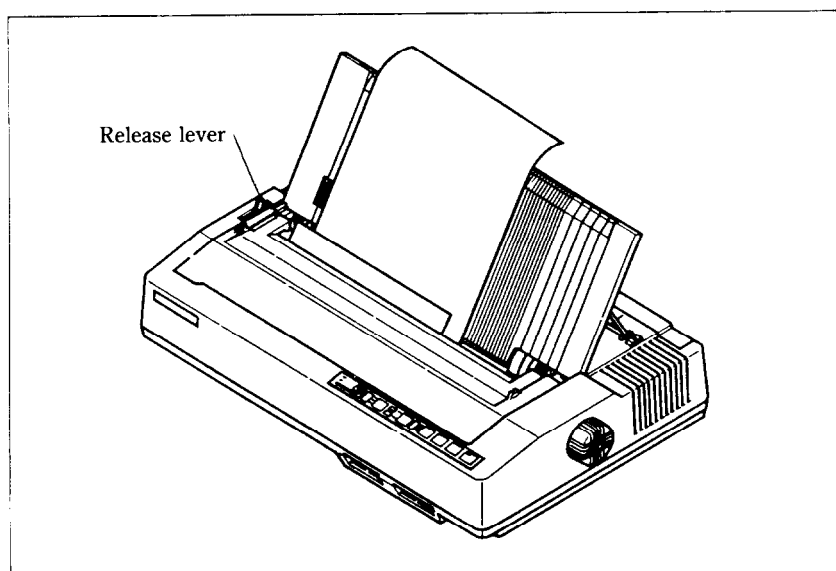
Figure 2-8. You can insert a sheet of paper automatically by using the release lever.

To align paper that is not in straight—
- Set the release lever to the adjustment setting.
- Straighten the paper and adjust it for the margins you want.
- Move the release lever back to its original setting.

Remember, you can also use the optional automatic sheet feeder for single sheets.

■ Loading sprocket-feed paper

This is standard computer paper, with the holes along the sides and perforations between the sheets, It is also called sprocket, punched, fan-fold, or just plain "computer paper". It can be as narrow as 4 inches, and up to 10 inches wide (4 inches to 15.5 inches on 15-inch type).

1. Turn on the printer and set the release lever for sprocket-feed paper.
2. If the paper guide is installed, remove it, and put it aside for a moment.
3. Place a stack of fan-fold paper behind the printer.
4. Open the sprocket covers, on the right and left sprocket units, as shown in Figure 2-9.
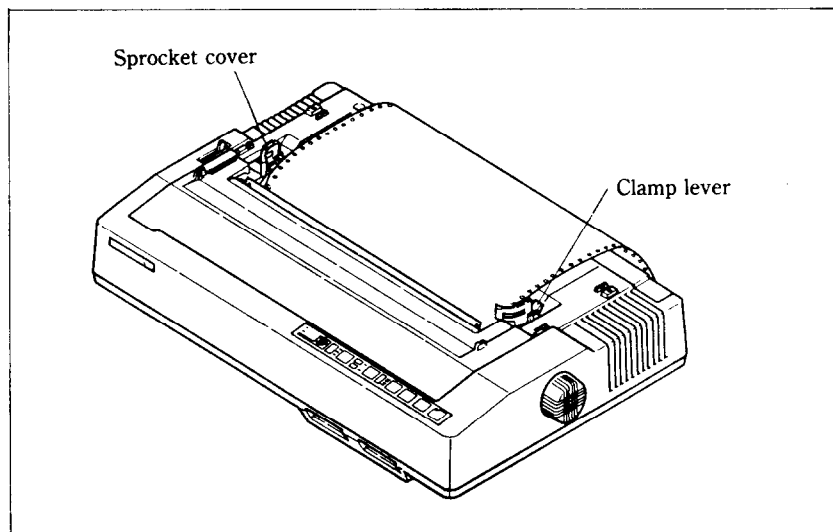
Figure 2-9. Open the sprocket covers to expose the sprocket teeth.

5. Flip the clamp levers backward. This allows the two sprocket units to move freely right and left so you can align them with the holes in the paper.
6. Feed the end of the paper into the slot next to the platen cover plate.
7. Fit the holes in the paper over the sprocket pins so the paper is even, and clamp the sprocket units in place.

8. Check that the paper is still positioned correctly, and close the sprocket covers (Figure 2-10).
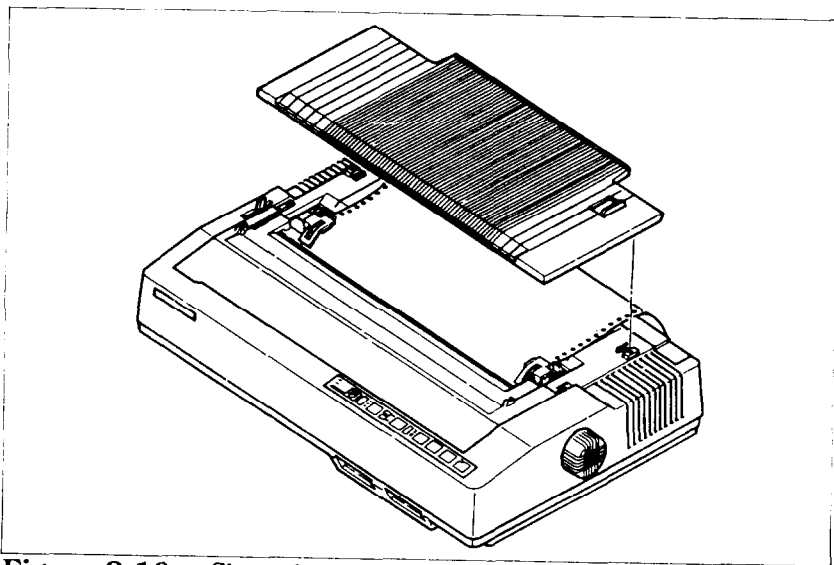
Figure 2-10. Close the sprocket covers when the paper is positioned correctly.
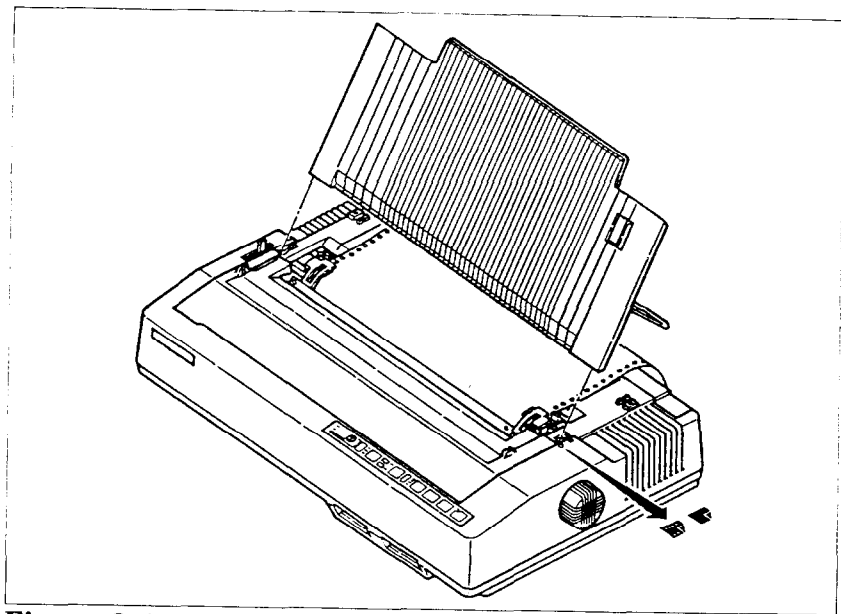
Figure 2-11. Ready to load the sprocket-feed paper. Note the reversed position of the paper guide.

9. Set the release lever to the auto-feed setting. The printer will advance the paper automatically.
10. When the paper stops, set the release lever for sprocket-feed paper again.
11. Install the paper guide on the printer. See Figures 2-10 and 2-11 for details. When used with sprocket-feed paper, the guide is turned around and laid flat (Figure 2-10) or raise (Figure 2-11) on top of the printer as a paper separator. Make sure that the tabs on the guide fit into their receivers.

ADJUSTING THE PRINT HEAD

The distance between the print head and the platen must be adjusted to accommodate papers of different thicknesses. To make this adjustment, move the adjustment lever, which is under the printer cover and immediately in front of the release lever (Figure 2-12). Pulling the adjustment lever towards you will widen the gap; pushing it away from you will narrow the gap.

There are four positions; you can feel the lever clicking into the various notches. The first step (shown) is the one most commonly used for single sheets of paper.

Try different positions until you get the best printing results.
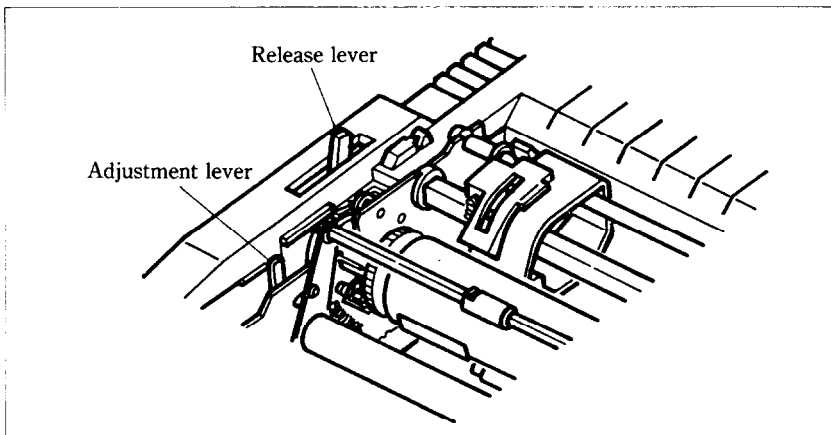
Figure 2-12. The adjustment lever allows for different thicknesses of paper.

TESTING THE PRINTER

Your printer has two built-in programs that printout sample lines of letters, numbers, and other characters — to show you that everything's in good working order. Self-test 1 is the long version, and self-test 2 is the short one. You can use either one to show the characters available in the printer, to check the installation of the ribbon and paper, and to check the adjustment of the head-to-platen gap.

You can print the self-test without hooking up the printer to your computer. Simply:

1. Plug in the printer (don't turn it on yet).
2. Insert a sheet of paper (or sprocket paper, either one).
3. While holding down one of the following keys, turn on the power switch:

> Paper Feed key — Self-test 1 (long)
> On Line key — Self-test 2 (short)

When you want to terminate the long self-test function, turn off the power switch.

CONNECTING THE PRINTER

Now that you've assembled and tested your printer, all that remains is to connect it to your computer as follows:

1. Make sure that both your computer and printer are turned off.
2. Connect the printer end of the interface cable to the connector socket at the right rear of the printer as shown in Figure 2-14.
3. Connect the other end of the cable to your computer as described in the computer manual.

Self-test 1

```
!"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcd
```

(printer self-test output lines)

Self-test 2

Standard mode

```
!"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcde
!"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdef
"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefg
#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefgh
$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghi
%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghij
```

IBM mode

```
!"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcde
!"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdef
"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefg
#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefgh
$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghi
%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghij
```

Figure 2-13. The self-tests show you what the printer can print.

Figure 2-14. Connect the cable like this.

```
efghijklmnopqrstuvwxyz{|}~ÇüéâäàåçêëèïîìÄÅÉæÆôöòûùÿÖÜ¢£¥₧ƒáíóúñÑªº¿
⌐¬½¼¡«»░▒▓│┤╡╢╖╕╣║╗╝╜╛┐└┴┬├─┼╞╟╚╔╩╦╠═╬╧╨╤╥╙╘╒╓╫╪┘┌█▄▌▐▀αßΓπΣσµτΦΘΩδ∞φε∩≡±≥≤⌠⌡÷≈°∙·√ⁿ²■
```

```
efghijklmnopqrstuvwxyz{|}~ÇüéâäàåçêëèïîìÄÅÉæÆôöòûùÿÖÜ¢£¥₧ƒáíóúñÑªº¿
```

```
efghijklmnopqrstuvwxyz{|}~ÇüéâäàåçêëèïîìÄÅÉæÆôöòûùÿÖÜ¢£¥₧ƒáíóúñÑªº¿
```

```
efghijklmnopqrstuvwxyz{|}~ÇüéâäàåçêëèïîìÄÅÉæÆôöòûùÿÖÜ¢£¥₧ƒáíóúñÑªº¿
```

```
stuvwxyz{|}~ÇüéâäàåçêëèïîìÄÅÉæÆôöòûùÿÖÜ¢£¥₧ƒáíóúñÑªº¿
```

```
stuvwxyz{|}~ÇüéâäàåçêëèïîìÄÅÉæÆôöòûùÿÖÜ¢£¥₧ƒáíóúñÑªº¿
```

```
efghijklmnopqrstuvwxyz{|}~ÇüéâäàåçêëèïîìÄÅÉæÆôöòûùÿÖÜ¢£¥₧ƒáíóúñÑªº¿
```

```
efghijklmnopqrstuvwxyz{|}~ÇüéâäàåçêëèïîìÄÅÉæÆôöòûùÿÖÜ¢£¥₧ƒáíóúñÑªº¿
```

```
fghijklmnopqrstuvwxyz{|}~ !"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGH
ghijklmnopqrstuvwxyz{|}~ !"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHI
hijklmnopqrstuvwxyz{|}~ !"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJ
ijklmnopqrstuvwxyz{|}~ !"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJK
jklmnopqrstuvwxyz{|}~ !"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKL
klmnopqrstuvwxyz{|}~ !"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLM
```

```
fghijklmnopqrstuvwxyz{|}~áíóúñÑªº¿⌐¬½¼¡«»
ghijklmnopqrstuvwxyz{|}~áíóúñÑªº¿⌐¬½¼¡«»
hijklmnopqrstuvwxyz{|}~áíóúñÑªº¿⌐¬½¼¡«»
ijklmnopqrstuvwxyz{|}~áíóúñÑªº¿⌐¬½¼¡«»
jklmnopqrstuvwxyz{|}~áíóúñÑªº¿⌐¬½¼¡«»
klmnopqrstuvwxyz{|}~áíóúñÑªº¿⌐¬½¼¡«»
```

MEMO

CHAPTER 3
USING THE PRINTER
WITH COMMERCIAL
SOFTWARE

Subjects covered in Chapter 3 include—
- **Using commercial software**
- **Controlling the printer with the Panel mode**

USING COMMERCIAL SOFTWARE

Many of you purchased this printer to use with commercial software. You made a good choice because this printer is compatible with most commercial programs, from word processing programs to spreadsheet programs to accounting programs.

Many of these programs have a routine for describing your printer. These routines are often in "installation programs". They typically give you a choice of printers or printer types to pick from. Some typical descriptions that you might pick for this printer are: "TTY type printer with backspace", "IBM-dot matrix printer", "Centronics-type printer", "Dot matrix ASCII printer" or "Epson LQ-1000". This printer should work fine with any of these descriptions.

Some printer lists are not very clear, and may not include anything that you think describes this printer. If you can't decide which description best fits this printer, we recommend that you narrow the list to two or three choices (you can quickly eliminate all the daisy-wheel printer types) and then experiment. You won't hurt anything if you guess wrong; it just won't work correctly. This should quickly tell you if your guess is right. If all else fails, though, your printer dealer will be happy to give you some advice.

Some programs don't ask you what kind of printer you have,

but instead ask some questions about what your printer can do. The answers to the "most asked" questions are: Yes — this printer can do a "backspace", and this printer can do a "hardware form feed".

With these questions answered, you are ready to start printing. Read the manual that came with your commercial software and the next Chapter to see how to make it send information for this printer to print. This is all you need to know to use this printer as a regular printer. But this printer isn't just a regular printer. This printer has many capabilities that your commercial software isn't aware of. We will see what it takes to use some of the printer's advanced features with commercial software in the next section.

WORD PROCESSING WITH THE PRINTER

Not many word processing programs directly support all of the advanced features of printers. They usually provide a method for using a few of the more common print features such as boldface and underlining. But as you are probably beginning to see from this manual, this printer can do much more than that.

As a result, most word processing programs provide a way of sending special codes to a printer. The actual codes used (as well as the method of entering them) will vary different software. The theory behind these methods, however, is basically the same.

This section discusses two word processing programs and one spreadsheet program most used by printer owners. The programs also provide a variety of ways to enter the codes necessary to use the advanced features of this printer. These concepts can be applied to many other programs besides those detailed here. The programs are:

    EasyWriter II
    WordStar
    Lotus 1-2-3

If your software program is not included in this Chapter, you should still study the different techniques used. Then, with the

help of your program manual and the supporting chapters in this manual, you should be able to figure out how yours works.

GENERAL CONCEPTS

Each word processing program has a way to get out of the standard text entry mode in order to accept the special printer function codes. WordStar uses the CONTROL key in different ways to define the print function codes.

EasyWriter II has a system function which allows you to define print pitches and special print functions for use with the ALT key.

Your word processing User Manual (if it supports this process) will have a section describing how to get out of the standard program. You will probably have to figure out on your own which codes are used. The general concepts and details of the two sample programs should be enough to help you be successful.

■ The escape code

Most of the special print functions start with a code called the escape code. It can be entered in decimal or hexadecimal values, by an ASCII character, or by using the control keys on your keyboard, depending on which program and which computer you are using.

The escape code tells the printer to interpret the values (or characters) following it as printer functions. The codes used to describe the functions are also entered in the same way as the escape code. In this section, we will show you the format each word processor uses as well as the general rules to correctly entering the function codes.

The next chapter covers how to convert the different forms of ASCII codes. You should review Chapter 4, if you have not already done so, before working with the function codes.

■ The master reset code

There is one function code which turns off all the print functions currently being used by the printer. It is called the master

reset code and resets the printer to its DIP switch settings. These print characteristics are the same as the ones used by the printer when it is first turned on.

The code sequence for master reset is ⟨ESC⟩ "@". By checking the ASCII equivalents in Appendix D, you can see that the decimal expression is 27 64. You'll see these numbers several times in this section.

Technically speaking, initializing the printer clears the print buffer, and the form length, character pitch, character set, line feed pitch and international character set are all reset to the values defined by their respective DIP switch settings.

We suggest you get in the habit of using the master reset code in any document where you use function codes. If you do not, the printer will keep the most recently defined characteristics, and print any following documents the same way.

You could turn your printer off each time (which also resets the default settings) but that would be hard on the printer circuits. Also, you'll save time and paper by letting the printer automatically reset with this code. (If you need more information on DIP switch settings for your printer, please refer to Appendix A.)

■ Using Letter Quality (LQ)

With letter quality, the printer prints more dots for each character than with the draft printing. This process results in a higher quality look to your text. Draft quality characters print much faster, so use them for your first drafts and use letter quality for a professional looking finished manuscript.

The escape code sequence to turn the LQ set on is ⟨ESC⟩ "x" 1 and the code sequence to select draft quality is ⟨ESC⟩ "x" 0. The decimal equivalents are 27 120 1 and 27 120 0, respectively.

■ Getting the most from your print choices

After working with the printer for a while, you may find that you want to add or change some of the print functions we have described in this chapter.

We suggest you do three things. First, you should review Chapter 4 and Appendix D to become as familiar as possible with ASCII codes and the function codes.

Second, read Chapter 4 which describes them in greater detail and shows examples of how they are used in BASIC programming. The functions will, for the most part, act the same in your word processing program. Understanding what's available and how they perform will help you use them correctly in your documents.

And third, follow the procedures in this chapter and your program User's Manual.

You may want to experiment with expanded text in combination with other print types. You can create some great-looking results with these functions. If you are unsure of any functions, review them first, then try some of your own samples.

USING THE PRINTER WITH EASYWRITER II

(Note: If you have not read the "General Concepts" section in this chapter, you should do so before continuing.)

This printer can be used with most of the standard print functions available with your EasyWriter II word processing program. These functions require no special adjustments to the printer or your program. They include:

1. Printing from the Print List Form screen.
2. Setting margins, tabs and lines per inch in the ruler line of your document. (The pitch settings, however, should be adjusted to obtain maximum use. They will be discussed later in this chapter.)
3. Print settings in the System Parameter function which are either default or new settings edited by you.

You can also redefine the print functions of EasyWriter II to take advantage of many of the printing capabilities of your printer. You may already be familiar with reconfiguring the printer driver from Appendix B of your EasyWriter II User's Manual. If not, don't be nervous; it's not as hard as it sounds. We will show you how to make changes in your program specifically to help you print with this printer.

By changing the pitch settings, you can use the document

ruler line to print pica, elite and condensed width pitches. In addition, you can use a similar method to print in letter quality.

The print control codes can be redefined to enhance the final product of your document. The boldface, underline, superscript and subscript functions require only a slight "recoding" of information in the printer driver. And we have some suggestions for changing the characteristics of the other print control codes to use italic, expanded, emphasized and italic-underline print. With these options, you will have even more flexibility printing with this printer.

■ Redefining pitch settings and print control codes

In order to change the settings used in the document ruler line and the print control codes, it is necessary to edit ASCII code decimal values in the System Functions portion of your EasyWriter II program. (For more details about ASCII codes, please refer to Chapter 4.)

Your printer is considered a Type B printer by the EasyWriter II program. Before making any changes in the printer driver, you should first check to be sure the printer selection is set for printer Type B (Option 7 on the System Functions menu.)

Then follow the instructions in Appendix B of your EasyWriter II User's Manual to reconfigure Type B printers. To become more familiar with the reconfiguration process and its terms and to make the instructions in this section easier to understand, we suggest you read through Appendix B first.

In these few paragraphs, we'll show you the ASCII decimal values we feel provide good flexibility in printing with this printer. You should follow the instructions hands-on with your own EasyWriter II program.

The changes you will make are for pitch settings and print control codes (also called font support). However, all the screens involved will be explained as you see them displayed.

From the System Functions Menu, choose Option 9 (Reconfigure Printer Type B) and the printer name will be displayed. Type over the present printer name as follows:

```
1. Printer Name  [Letter Quality Printer]
```

Press RETURN and the Edit Global Sequences screen will be displayed. These codes control the print functions for form feed, line feed, margin settings and automatic justification. We do not recommend that you edit any of these codes.

Press RETURN and the Edit Pitch Table screen will be displayed. On this screen, you will enter the ASCII decimal values to define the print pitches. The first two fields in each line define the pitch range (which in this case are both the same number). They should be assigned as follows:

```
10 = Pica              1 = Letter Quality On
12 = Elite             2 = Letter Quality Off
17 = Condensed Pica    3 = Master Reset Code
```

On this screen, the column labeled "Sequence" is used to define the print functions in their ASCII decimal values. For these print pitches, we will use a combination of codes to turn letter quality on and off and to choose the function code for each pitch. (For more details on function codes, please refer to Appendix D.)

Follow the sample and enter the (italic) codes for lines 17 - 22.

```
17. [10 ] [10 ]        [ 18   27  80                        ]
18. [12 ] [12 ]        [ 18   27  77                        ]
19. [17 ] [17 ]        [ 15                                 ]
20. [1  ] [1  ]        [ 27  120    1                       ]
21. [2  ] [2  ]        [ 27  120    0                       ]
22. [3  ] [3  ]        [ 27   64                            ]
23. [120] [120]        [                                    ]
24. [120] [120]        [                                    ]
25. [120] [120]        [                                    ]
26. [120] [120]        [                                    ]
```

The codes 120 in lines 23 through 26 can be changed to reflect more pitch settings. We recommend that until you are more familiar with using special function codes, you use just the six we have defined.

When you have finished, press RETURN. You will be

transferred to the Edit Line Spacing screen. Do not change these codes. They define how many lines per inch the printer uses. Press RETURN to transfer to the Edit Font Support screen.

Change all of the entry fields to Option 2 (Control Code Support) on the Edit Font Support screen. Also, make changes in the other fields to look like the figure shown below. Enter the (italic) codes for lines 41 - 50.

```
41. Bold/Shadow Face Support   [2]
42. Single Underline Support   [2]   Using Character [95 ]
43. Double Underline Support   [2]   Using Character [0  ]
44. Overstrike Support         [2]   Using Character from
45. Special (Color) Support    [2]   System Parameters
46. Sub/Superscript Support    [2]
47. Will underline retain font (Y) or be normal font (N)?
    [N]
49. Start double underline [    ]
50. After double underline [    ]
```

When you have finished, press RETURN and the Edit Font Sequences screen will be displayed. Here you will define print control codes for use in your documents. As with the pitch settings, ASCII decimal values are used that correspond to the print function assigned to each control key. Table 3-1 shows the current control function, the print function we will assign to it and the keyboard keys used.

Table 3-1
EasyWriter II control keys

EasyWriter II Function	New function	Keys used
Boldface	Boldface	ALT & B
Shadow	Italic	ALT & S
Underline	Underline	ALT & −
Double	Expanded	ALT & =
Overstrike	Emphasized	ALT & O
Special	Italic Underline	ALT & *
Subscript	Subscript	ALT & D
Superscript	Superscript	ALT & U

Enter the (italic) codes for lines 51 - 66.

51. Normal to Bold	[	*27*	*71*			]
52. Bold to Normal	[	*27*	*72*			]
53. Normal to Shadow	[	*27*	*52*			]
54. Shadow to Normal	[	*27*	*53*			]
55. Normal to Underline	[	*27*	*45*	*1*		]
56. Underline to Normal	[	*27*	*45*	*0*		]
57. Normal to Double	[	*27*	*87*	*1*		]
58. Double to Normal	[	*27*	*87*	*0*		]
59. Normal to Overstrike	[	*27*	*69*			]
60. Overstrike to Normal	[	*27*	*70*			]
61. Normal to Special	[	*27*	*52*	*27*	*45*	*1*]
62. Special to Normal	[	*27*	*53*	*27*	*45*	*0*]
63. Normal to Subscript	[	*27*	*83*	*1*		]
64. Subscript to Normal	[	*27*	*84*			]
65. Normal to Superscript	[	*27*	*83*	*0*		]
66. Superscript to Normal	[	*27*	*84*			]

When you have finished, press RETURN. You're done! You will be transferred out of the Reconfigure Type B Printer function and back to the System Functions Menu.

■ A sample printout with EasyWriter II

Let's look at a short example to demonstrate how pitch settings and print control keys can be used in a document. The example below shows the use of expanded and italic prints used in combination with condensed and pica pitch settings. Use your EasyWriter II program hands-on and type the example below.

```
SUBJECT: ORDERING STATIONERY SUPPLIES
I would like to place an order for stationery supplies
from your mail order catalog. Enclosed is my order form
and a check for $247.67. Please process this order as
soon as possible. Thank you.
```

With the cursor under the "S" in "SUBJECT", set the print pitch in the ruler line to condensed width pitch. Name the ruler line "condensd" (without the quotes) and change the character pitch to 17 and the line spacing to 6. To make the subject title expanded, use the ALT and = keys (in the line mode) to highlight the line.

Now, change the pitch setting in the next line to pica by set-ting a new ruler line: Ruler Name — pica; Character Pitch — 10. Use the print control key S (for italic) to highlight the second sentence in the paragraph. Move the cursor to the "E" in "Enclosed" and (in the sentence mode) use the ALT and S keys to highlight the sentence. (You'll have to press S twice to get the . 67.)

At the end of the document, reinitialize the printer to its default settings with a new ruler line using the Master Reset code. Ruler Name — reset; Character Pitch — 3.

Print the document. Your printout should look like this:

```
SUBJECT: ORDERING STATIONERY SUPPLIES

    I would like to place an order for stationery supplies
from your mail order catalog. Enclosed is my order form
and a check for $247.67. Please process this order as
soon as possible. Thank you.
```

The subject title will print in expanded condensed characters which are twice the width as standard condensed characters. The sentence in the paragraph is printed in italic pica print. The last ruler line will reinitialize the printer. (See the general con-cepts section for more details on master reset.) This is just one example; you should be able to apply most of the function codes to the setup used here.

■ Redefining your own print pitches

If you want to define a new print pitch (Edit Pitch Table), enter your function code to choose the print you want.

For example, if you find yourself frequently using italic print for large blocks of text in pica pitch, you can combine italic and pica pitch to define italic pica and use it in the ruler line of your document. The ASCII code sequence would be 27 80 27 52 to print italic pica pitch.

■ Redefining your own print control keys

The ASCII codes to redefine the print control keys (Edit Font Sequences) are pretty straightforward. There are individual

ASCII decimal values to turn on and off different prints. You want to affect that aspect but not the print pitch itself. Leave that for your document ruler line. Remember, all the codes can be found in Appendix D of this manual.

Also, keep in mind that print control keys can be combined in your document such as boldface and underline. EasyWriter II uses three methods of highlighting on the display screen. It highlight, underlines and shows reverse image characters. You cannot combine print control functions that use the same method of highlighting.

For example, in our definitions, underline and expanded prints are both displayed as underlined on the screen. Whichever function you use last will cancel out any previous modes.

USING THE PRINTER WITH WORDSTAR

(Note: If you have not read the "General Concepts" section in this chapter, you should do so before continuing.)

This printer supports many of the standard WordStar printing capabilities without requiring any changes. You can:

1. Print documents from the No-File Menu.
2. Use dot commands except for lines per inch and characters per inch.
3. Print boldface, underline, double-strike, strikeout, superscript and subscript characters as well as use print pause.
4. Select and print variable information for marge letters, etc.

■ User-defined print commands

There are several CONTROL-P (^ P) commands that automatically work with this printer and require no changes. They include:

^ PS Underscore   ^ PB Boldface
^ PD Double-strike   ^ PX Strikeout
^ PT Superscript   ^ PV Subscript
^ PC Print Pause

It is also possible to define the ^PA (alternate pitch) command to change the print pitch of your document. The WordStar User's Manual fully describes the use of these ^P functions. You should refer to your manual if you need help with them.

There are four alternate ^P codes that can be defined during the installation of your WordStar program to perform other printer functions. They are ^PQ, ^PW, ^PE, and ^PR. The process of defining ^P commands is called "patching" and is a fairly complicated process. Once you have successfully defined these codes, they are inserted in your text exactly like other ^P commands. If you wish to use them, refer to the WordStar User's Manual for instructions or contact your dealer for assistance.

Perhaps the most useful user-defined ^P command is ^PE. If you define this as an escape (ASCII code 27), you can then access nearly all of the advanced features of this printer. Without this patch, you cannot place an escape in the WordStar document and subsequently, you are limited to using WordStar's repertoire of print functions. A shame when you have a powerful printer!

USING THE PRINTER WITH LOTUS 1-2-3

Lotus is one of the integrated software packages that include a spreadsheet, a database manager and graphics. We will look at how to print Lotus 1-2-3 spreadsheet.

Lotus 1-2-3 uses the /Print command to print spreadsheets. When you enter /P, a menu appears that presents you with a number of choices. Lotus 1-2-3 gives you a lot of flexibility in printing spreadsheets through this menu, but you *have* to define the range to print. All the other items have default values that make getting started easy.

If you do change several of the things listed in the /Print menu, Lotus 1-2-3 will remember the selections that you have made and use them each time you print the spreadsheet. They are even saved with the spreadsheet so that they will be the same the next time that you use the spreadsheet.

You can specify the range to print in all the normal ways: by pointing, by typing the cell addresses of the endpoints, by entering a range name, or by using the F3 key to point to a range name.

After you have specified a range to print, and changing any of the other options that you wish, start printing the spreadsheet by selecting the *Go* option. Lotus 1-2-3 will split the spreadsheet into sections to fit onto pages if it won't all fit on one page.

Let's look at some of the other options on the */Print* menu, and see how they add to the flexibility of printing spreadsheets.

The *Line* option advances the paper one line. Use this to put space between different sections of your spreadsheets when you print them. The *Page* option advances the paper to the top of a new page. Use this option to start on a new page.

Selecting the *Align* option tells Lotus 1-2-3 that you have moved the paper to the top of a new page. Use this option after using the Paper Feed key to move the paper or after inserting a new single sheet of paper.

The *Clear* option allows you to clear any or all of the other options that you have selected. The *Quit* option ends the */Print* command and returns you to Ready Mode.

Selecting *Options* from the */Print* menu presents you with some additional page format selections.

You can add *Headers* or *Footers* to each page of your output. A header is a line that prints at the top of each page, while a footer is a line that prints at the bottom of each page.

Lotus 1-2-3 has three characters that perform special functions when they are included in a header or a footer. You can include sequential page numbers on each page by including the # character where you want the page number to print (For example, Page #).

The current date will be printed if you include the @ character in a header or footer. (For example: As of @.)

You can direct sections of headers and footers to the left, right, or center by using the | character. Each header or footer is divided into three sections; Left, center, and right. The | character shows the limits of these sections. So to print a header with the date to the left, a title in the center, and a page number to the right, the header might look like this:

@ : Spreadsheet Title : Page #

And, on August 12, 1986, the results might look like this:

12-Aug-86   Spreadsheet Title   Page 1

Another of the selections under *Options* is *Setup*. This selection allows you to create a *setup string* that will be sent to the printer before each section of the spreadsheet is printed. You can include non-printing codes in the setup string by using a backslash (\) followed by a three digit number that consists of the decimal ASCII value for the code that you wish to send (with leading zeros if required). For example, to print a worksheet in condensed print, use the setup string \015. This sends ASCII 15 which is the code for condensed printing. The following table shows how many character columns will fit with different printing width, and the setup string to get each width.

Table 3-2
Print columns on a page with Lotus 1-2-3

Pitch	Characters per line		Setup codes
	10-inch type	15-inch type	
Pica	80	136	\018\027P
Elite	96	163	\018\027M
Condensed Pica	137	233	\027P\015
Condensed Elite	160	272	\027M\015
Expanded on	—	—	\027W\001
Expanded off	—	—	\027W\000
LQ on	—	—	\027x\001
LQ off	—	—	\027x\000
Proportional on	—	—	\027p\001
Proportional off	—	—	\027p\000

USING THE PANEL MODES

At the end of this chapter, we'll explain about the "Panel" mode, which is powerful function of the printer. Some commercial software does not support defining your own selections. In

this case, use the "Panel" modes.

Selection of any of the Panel Modes at power-on ensures that your choices remain in effect until you turn the power off. This means that the printer will ignore any codes sent by the software that you are using. The Panel Modes allow selection of the following printing attributes:

Key	Type of Panel mode
Type Style	Type Style Panel mode
Quality	Quality Panel mode
Print Pitch	Print Pitch Panel mode
Top Of Form	All Panel mode
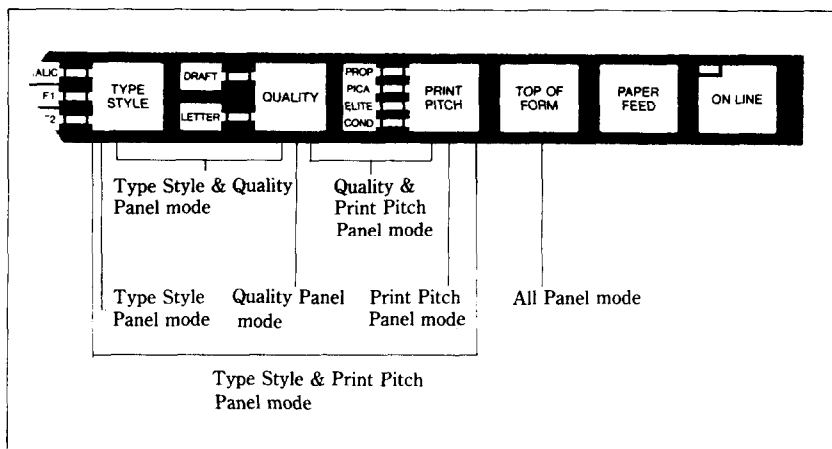
Figure 3-1. The Panel Modes — hold down keys while turning power on.

If you want to set only one attribute, for example, the Print Pitch Panel mode, press the PRINT PITCH key on the control panel when you turn on the power. After a second, you will hear a beep, and the ON LINE indicator will glow. To change the pitch to another, press the ON LINE key, select the desired pitch, and press the ON LINE key again. The Print Pitch Panel mode is now set to the pitch of your choice.

Similarly with Draft and LQ — turn on the power while holding the QUALITY key, and wait for the ON LINE indicator glow. If you want to change the character, press the ON LINE

key, press the QUALITY key once again to make your choice, and press the ON LINE key again.

To select the Type Style Panel mode, press the TYPE STYLE key while turning on the power. To change the character set to another, press the ON LINE key, select the desired type, and press the ON LINE key again. The Type Style Panel mode is now set to the character set of your choice.

More than one attribute may be set in this way. For example, if you wish to set both print pitch and print quality, press *both* the PRINT PITCH and QUALITY keys when turning on the power, and after the ON LINE indicator glows, make further selections as necessary.

If you wish to set all three panel modes, you can use the TOP OF FORM key instead of using the three keys.

ADVANCED PANEL FUNCTIONS

The printer has the capability to do some basic formatting from the control panel, as well as move the platen by precise amounts. The following formatting and platen movements may be performed from the control panel.

Key	Functions
On Line & Paper Feed	Forward micro-feed
On Line & Top Of Form	Reverse micro-feed
Print Pitch & Top Of Form	Top of form setting
Quality & Top Of Form	Setting of left margin
Quality & Paper Feed	Setting of right margin
On Line & Type Style	Clear the print buffer

■ Setting print start position

When you want to align the print start position, you can set it by the micro-feed operation with the control panel, instead of turning the platen knob manually.

1. Set the Off Line mode by pressing the On Line key.
2. While holding down the On Line key, press one of the following keys.
   Paper Feed key — Forward micro-feed
   Top Of Form key — Reverse micro-feed
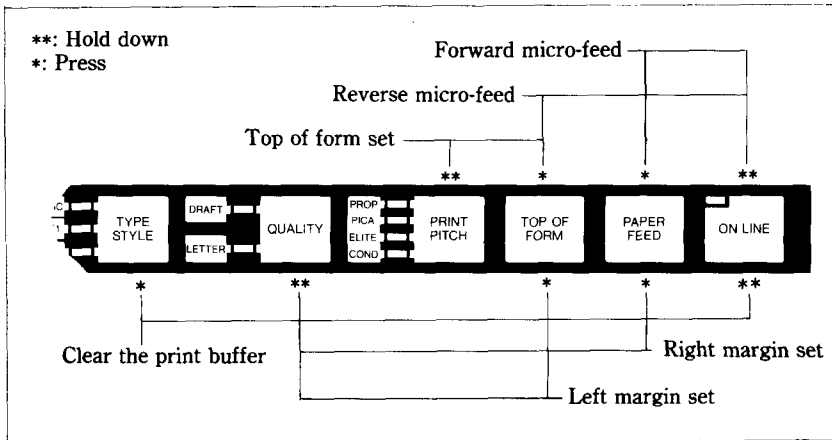
Figure 3-2. You can set many functions by the combinations of the control panel keys in the Off Line mode.

3. When you can set the print start position, release the Paper Feed key or the Top Of Form key first, then release the On Line key.

■ Setting the top of form
 When you turn on the printer, the top of form is automatically set to the current position. If you want to change the position, you can reset it by the following procedures.

1. Set the Off Line mode by pressing the On Line key.
2. While holding the Print Pitch key, press the Top Of Form key. Your printer acknowledges the new top of form with the sound of beep.

■ Setting the left and right margins
 As you'll learn in Chapter 5, you can set the left and right margins with the control codes. In addition, you can set them manually just like a typewriter by the following procedures.

1. Set the Off Line mode by pressing the On Line key.
2. While holding the Quality key, press one of the following keys.
  Top Of Form key — Left margin set

Paper Feed key — Right margin set

3. While holding the two keys, the print head moves across the page step-by-step.
4. When the print head goes to the position where you want to set margin, release the two keys. So the printer acknowledges the margin with the sound of beep.

■ Clearing the print buffer

You can clear the contents of the print buffer with the control panel by the following procedures, instead of turning off the printer.

1. Set the Off Line mode by pressing the On Line key.
2. While holding the On Line key, press the Type Style key.
3. While holding the two keys more than three seconds, the contents of the print buffer are cleared with the sound of beep.

CHAPTER 4

PRINTING WITH BASIC

Subjects covered in Chapter 4 include—
- **Listing BASIC programs on the printer**
- **How a program prints things**
- **Control codes, escape codes, and command syntax**
- **Selecting the right software mode**
- **Letter quality (LQ) characters**
- **Fixed and proportional character spacing**
- **Special printing—**
    Printing in italics
    Underlining and overlining
    Superscripts and subscripts
    Boldface and emphasized text
    Mixing print modes

To show you how to control your printer from a program, we have chosen BASIC because it is easy to learn and easy to use. Also, more personal computer users program in BASIC than in any other language.

The rest of this manual will show you a little BASIC — just enough for you to use your printer. We're not going to try to make you an expert programmer, though, only get you started. There are many excellent books that will teach you BASIC, so if you discover that you like to program you should have no trouble learning more about it.

SOME BASICS OF BASIC

■ First steps

The first things that a beginner learns to do are to list a program and to print a character string. Certainly these are the easiest operations one can do, but even they may depend on what computer you have. In Microsoft BASIC, we can list all the steps in a program by entering LIST. This lists them on the CRT screen; if we want to print them on a printer, we prefix the command with an L (enter LLIST).

The Microsoft BASIC command for outputting information is PRINT. Like the LIST command, this displays the information on the CRT screen so we have to add an L (→ LPRINT) if we want to use the printer. Just put whatever you want to print between quotes and after LPRINT (anything enclosed in quotes is called a character string). For example, we would use LPRINT "Hello!" to output "Hello!" to the printer. We'll see later how to LPRINT more than just character strings.

We started with Microsoft BASIC because it is the most widely used version of BASIC. The programs in this manual are written in Microsoft BASIC so they should run on most computers. But if strange things happen when you try to run a program, check the BASIC manual that came with your computer.

Let's consider Apple II computers for a minute. These popular computers use their own brand of BASIC. To use an Apple II, enter the following —

```
PR#1           PR#1
LIST           PRINT "Hello!"
PR#0           PR#0
```

The PR#1 tells the Apple to send everything to the printer, the LIST or PRINT command sends it, and the PR#0 returns output to the screen.

Now that we know how to address the printer, let's try listing a BASIC program. We will load a program into memory ready to program printer operation — just as soon as we learn a little bit about the ASCII codes.

■ ASCII codes and the CHR$ function

You can talk to your computer in BASIC, but your computer and your printer talk to each other in what are known as ASCII codes. In the ASCII code, each number from 0 to 255 has a particular meaning — 36, for example, makes the printer print a dollar sign. Some numbers cause the printer to do other things, too. For instance, sending a 7 sounds the printer's bell.

Taken together, these numbers and their meanings make up the ASCII code (pronounced *ask-key*), which stands for the *American Standard Code for Information Interchange*. There are ASCII codes for all the letters of the alphabet (upper case and lower case), 0 to 9, most punctuation marks, and some (but not all) of the functions of the printer.

There are a number of different ways to represent an ASCII code, depending on how you are using it. For example, the ASCII codes for the upper-case letter "A" are 65 (decimal) or &H41 (hexadecimal). Or you can just call it "A". Appendix C shows all of the ASCII codes.

BASIC uses the CHR$ function to represent ASCII characters and many functions. To print the letter "A" we would enter LPRINT CHR$(65). To make the printer's bell sound, we would enter LPRINT CHR$(7). In general, we print a character by entering LPRINT CHR$(*ASCII code*) to the printer.

We can also use hex ASCII codes. Although we use only decimal ASCII codes in this manual, in some applications it will help if you understand what a hex code is. "Hex" is short for hexadecimal and refers to a base-16 number (the numbers we use in everyday life are base 10). Since the hex system needs 16 digits, it uses the numerals 0 through 9 and also the letters A through F. In BASIC programs, you can always tell that a number is in hexadecimal by the "&H" immediately preceding it. The ASCII code for the letter "A" (65 in decimal) is &H41 in hex.

■ Control codes

ASCII codes with values of 32 or less do not correspond to keys on the keyboard. These codes control many of the printer's functions, so we call them control codes. To enter a control code from the keyboard, we have to press two keys at the same time

— the "control" key and one other. The other key determines
what code is sent — pressing the control (CTRL) key and A
sends ASCII code 1, CTRL B sends ASCII code 2, and so on.

Your printer has a lot of control codes to let you do some
useful things. Let's try one that we've mentioned several times
already:

```
10 ' Demo of ASCII code
20 LPRINT CHR$(7)
30 END
RUN
```

That's the printer's bell (we call it that even though it sounds
like a buzzer). We'll learn more about it in later — we just
wanted to show you a control code that would get your attention
right away!

There are four common ways of referring to a control code:
the name of the code or its abbreviation, the decimal ASCII
value, the hexadecimal ASCII value, and the "CTRL-" value.
For example, the ASCII code that causes the printer to advance
the paper one line is decimal 10. This code may be referred to
by any of the following:

line feed	— the name of the code
⟨LF⟩	— its abbreviation
ASCII 10	— its decimal value
ASCII &H0A	— its hexadecimal value (the &H signifies hex)
CHR$(10)	— the way it's used in BASIC
CTRL-J	— the way you send it from a keyboard

Of course, most of the time we don't need to bother with
these. The computer is smart enough to know that when we
press the "A" key we want to print the letter "A" — it takes
care of all the intermediate steps.

Appendix C is a table that shows the various names for each
code so you can convert back and forth. The microcomputer
world is not very consistent in describing ASCII codes, so it's
important that you have a basic knowledge of them.

■ The escape codes

Back when the ASCII system was set up, computer equipment was relatively simple and thirty-three control codes were considered sufficient at the time. The American Standards people realized that eventually more control codes would be needed, so they included the escape (ESC) code to allow almost any number of additional codes to be defined when they became necessary.

ESC allows us to "escape" from the ordinary set of control codes so we can specify additional functions and other information needed for a printer function. In this manual, we'll write the ESC code inside broken brackets, like this — ⟨ESC⟩.

⟨ESC⟩ — decimal 27 — is always followed by at least one other number; it is never used alone. The whole series of related numbers is called an escape sequence.

■ A note on command syntax

Because the readers of this manual will be running such a wide variety of applications on so many different computers, we can't show the exact way of sending codes to the printer for each one of them. Instead, as we introduce you to each new command, we will show the commands as in this example:

⟨ESC⟩ "W" 1

This command turns on expanded printing. ⟨ESC⟩, as we mentioned earlier, is the escape code (which is ASCII code 27). A letter or number in quotes (such as the "W" above) means that the character should be sent to the printer, but without the quotes. In our example, you should send a capital W following the escape code. In BASIC, you could do this in a couple of ways: by sending the character itself (e.g. LPRINT "W";), or by using the CHR$ function to send the ASCII code for the character (e.g. LPRINT CHR$(87);).

Many of printer commands end with a 1 or 0. When shown as in the above example (i.e., no quotes and no "CHR$"), you can use either ASCII code 1 (i.e., CHR$(1)) or the character "1" (which is ASCII code 49). The same principle applies to commands ending with 0.

So for our example above, any of these BASIC statements will have the same result:

```
LPRINT CHR$(27);CHR$(87);CHR$(1)
LPRINT CHR$(27);CHR$(87);CHR$(49)
LPRINT CHR$(27);"W";CHR$(1)
LPRINT CHR$(27);"W";CHR$(49)
LPRINT CHR$(27);"W1"
```

Even though there are many commands that require the use of ASCII code 0 (the ⟨NUL⟩ character), the number "0" (zero or ASCII code 48) cannot be substituted. In these cases, instead of an unadorned 0 we will show CHR$(0) each time these commands are referenced.

That's it for the basics. You are now ready to learn how to use the many features of your printer.

■ Selecting the right software mode

For this printer to correctly respond to control codes, you must make sure that you've set the DIP switches properly.

The rest of this manual discuss the various control codes to control this printer. You'll find that many features have different codes for either the Standard mode or the IBM modes.

The Standard Mode emulates the Epson LQ-1000 printer. The IBM-G Mode emulates the IBM Graphics printer and the IBM-P Mode emulates the IBM Proprinter. You can select one of these modes by setting DIP switches 2-1 and 2-2. (For details, please refer to Appendix A.)

SOME SPECIAL KINDS OF TEXT

If you looked carefully at your printer's self test, you probably noticed that it can print in italics. But that's not all! Your printer can underline or overline characters, print superscripts and subscripts, and perhaps most exciting, print letter quality characters.

■ Draft and Letter Quality characters

This printer's Letter Quality (sometimes abbreviated as LQ) character set is ideal for correspondence and other important printing, as it produces very high quality characters. In addition, your printer prints draft quality characters. This is adequate for most work and has the fastest printing speed. For the final printout, try Letter Quality. The program below shows an example.

```
10 ' Demo of Draft and LQ characters
20 LPRINT CHR$(27);"x0";
30 LPRINT "This line shows DRAFT characters."
40 LPRINT CHR$(27);"x1";
50 LPRINT "This line shows LETTER QUALITY!"
```

In this program, line 20 selects draft characters with ⟨ESC⟩ "x" 0 command. Line 30 prints a sample before line 40 switches printer back to LQ characters with an ⟨ESC⟩ "x" 1. When you run the program you should get this:

This line shows DRAFT characters.
This line shows LETTER QUALITY!

Now turn off the power switch. After you turn on the power switch while holding the Quality key down on the control panel, set the LQ mode with the Quality key while in the Off Line mode. Then run the program again. You should get like this:

This line shows DRAFT characters.
This line shows LETTER QUALITY!

As you can see, now the printout only contains LQ characters. This is because we set the printer to the "Quality Panel" mode at power on, so the printer ignored the control codes to set or cancel LQ characters. This is a very useful way to print LQ characters if you are using a packaged software which does not support the LQ character mode.

Table 4-1
Letter quality commands

Function	Control code
Letter quality ON	⟨ESC⟩ "x" 1
	⟨ESC⟩ "I" 2 (IBM-P mode only)
Letter quality OFF	⟨ESC⟩ "x" 0
	⟨ESC⟩ "I" 0 (IBM-P mode only)

■ Italic printing

Italic letters are letters that are slanted to the right. Your printer can print all of its letters in *italic* as well as the roman (standard) letters you are accustomed to. Italics can be used to give extra emphasis to certain words. The commands to turn italic on and off are shown in Table 4-2.

Table 4-2
Italic commands

Function	Control code
Italic ON	⟨ESC⟩ "4" (except IBM-P mode)
	⟨FS⟩ "4" (IBM-P mode only)
Italic OFF	⟨ESC⟩ "5" (except IBM-P mode)
	⟨FS⟩ "5" (IBM-P mode only)

Use this program to see italic characters:

```
10 ' Demo of italic and roman
20 LPRINT CHR$(27);"4";
30 LPRINT "This line is in ITALIC characters."
40 LPRINT CHR$(27);"5";
50 LPRINT "This line is in ROMAN characters."
```

Here is what you should get:

This line is in ITALIC characters.
This line is in ROMAN characters.

In this program, line 20 turns italic on with ⟨ESC⟩ "4", and line 40 turns italic off with ⟨ESC⟩ "5".

Now, set the "Type Style Panel" mode, as described in

Chapter 3, and set the Italic type, then run the program again. You should get like this:

This line is in ITALIC characters.
This line is in ROMAN characters.

As you can see, this time the printer ignored the control codes to set or cancel italic characters.

■ Characters in the Font Cartridge

This printer offers you the Prestige LQ characters as standard. If you want to use other LQ character styles, use the optional font cartridges.

You can install those font cartridges following the procedures described in Chapter 1. You can control the characters in those font cartridges as shown below:

```
10 ' Demo of optional characters
20 LPRINT CHR$(27);"k";CHR$(1);
30 LPRINT "This line shows Font 1 characters."
40 LPRINT CHR$(27);"k";CHR$(2);
50 LPRINT "This line shows Font 2 characters."
60 LPRINT CHR$(27);"k";CHR$(0);
70 LPRINT "This line shows Internal Characters."
```

When you've installed the Gothic Font Cartridge to the font slot 1, and the Orator Font Cartridge to the font slot 2 with the 15-inch type, the results of this program look like this:

This line shows Font 1 characters.
THIS LINE SHOWS FONT 2 CHARACTERS.
This line shows Internal Characters.

In this program, line 30 selects the characters on the font slot 1, with ⟨ESC⟩ "k" CHR$(1) command. Line 40 prints a sample with the LQ characters in the font cartridge installed on the font slot 1. Line 50 selects the font cartridge slot 2 with ⟨ESC⟩ "k" CHR$(2) command. Line 60 prints a sample with the font slot 2

characters before line 70 switches printer back to internal LQ characters with an ⟨ESC⟩ "k" CHR$(0) command.

You can control the font styles with the extra functions with the control panel.

Table 4-3
Font selection commands

Function	Control code
Selects font slot 1	⟨ESC⟩ "k" CHR$(1)
Selects font slot 2 (15-inch type only)	⟨ESC⟩ "k" CHR$(2)
Selects internal characters	⟨ESC⟩ "k" CHR$(0)

■ Underlining and overlining

Not only can your printer print all styles of printing in both roman and italic, but it can underline and overline them too. The control codes are shown in Table 4-4.

Table 4-4
Underline and overline commands

Function	Control code
Underline ON	⟨ESC⟩ "−" 1
Underline OFF	⟨ESC⟩ "−" 0
Overline ON	⟨ESC⟩ "_" 1
Overline OFF	⟨ESC⟩ "_" 0

It's that simple. Let's try it with this program:

```
10 ' Demo of underlining and overlining
20 LPRINT "Demo of ";
30 LPRINT CHR$(27);"-1";
40 LPRINT "UNDERLINED";
50 LPRINT CHR$(27);"-0";
60 LPRINT " and ";
70 LPRINT CHR$(27);"_1";
80 LPRINT "OVERLINED";
90 LPRINT CHR$(27);"_0";
100 LPRINT " printing."
```

It should come out like this:

Demo of U̲N̲D̲E̲R̲L̲I̲N̲E̲D̲ and O̅V̅E̅R̅L̅I̅N̅E̅D̅ printing.

In this program underline is turned on in line 30 with ⟨ESC⟩ "—" 1, and then off in line 50 with ⟨ESC⟩ "—" 0. The overline is turned on in line 70 with ⟨ESC⟩ "_" 1, and then off in line 90 with ⟨ESC⟩ "_" 0. There's a new aspect to this program, though. The semicolons at the end of the lines told BASIC that those lines were to be continued. Therefore, BASIC didn't send a carriage return and line feed at the end of those lines. We just did this to illustrate that all these control codes can be used in the middle of a line. It's easy to underline, overline or *italicize* only part of a line.

■ Superscripts and subscripts

Your printer can print in two different heights of characters. The smaller characters are called *superscripts* and *subscripts* and are half the height of normal characters. *Superscripts* print evenly with the tops of regular printing while *subscripts* print evenly with the bottom of regular printing. They are frequently used to reference footnotes, and in mathematical formulas.

Table 4-5 has the codes for using superscripts and subscripts.

Table 4-5
Superscripts and subscripts commands

Function	Control code
Superscript ON	⟨ESC⟩ "S" 0
Subscript ON	⟨ESC⟩ "S" 1
Super and subscript OFF	⟨ESC⟩ "T"

Try this program to see them work:

```
10 ' Demo of superscripts and subscripts
20 LPRINT "Look! ";
30 LPRINT CHR$(27);"S0";
40 LPRINT "SUPERSCRIPTS ";
50 LPRINT CHR$(27);"T";
60 LPRINT "& ";
70 LPRINT CHR$(27);"S1";
80 LPRINT "SUBSCRIPTS ";
90 LPRINT CHR$(27);"T";
100 LPRINT "on one line."
```

Look! SUPERSCRIPTS & ₛᵤᵦₛᶜᵣᵢₚₜₛ on one line.

Here line 30 turns on superscripts with ⟨ESC⟩ "S" 0. It's
turned off in line 50 with ⟨ESC⟩ "T". Then between printing
text, subscripts are turned on in line 70 with ⟨ESC⟩ "S" 1, and
finally off in line 90. Again, everything prints on one line
because of the semicolons.

CHANGING THE PRINT PITCH

In "printer talk," the number of characters that can be printed
in one inch is called the print pitch or character pitch. Normally,
your printer is set for 10 characters per inch, which is called *pica*
(and is the same as the pica pitch on most typewriters). This
works out to 80 characters per line for the 10-inch type, or 136
characters per line for the 15-inch type.

You can also print 12 character per inch (*elite pitch*) which
gives you 96 or 163 characters per line, or 15 character per inch
(*semi-condensed pitch*) which gives you 120 or 204 characters per
line.

You can set these pitches by using the Print Pitch key on the
control panel manually, or by software as shown in the table
below.

Table 4-6
Print pitch commands

Pitch	Characters/inch	Control code
Pica	10	⟨ESC⟩ "P"
Elite	12	⟨ESC⟩ "M"
		⟨ESC⟩":"(IBM-P mode only)
Semi-condensed	15	⟨ESC⟩"g"(except IBM-P mode)

Try this program to see how these three pitches work. Be sure
to set the printer to Standard mode.

```
10 ' Demo of print pitches
20 LPRINT CHR$(27);"g";
30 LPRINT "This line is SEMI-CONDENSED pitch."
40 LPRINT CHR$(27);"M";
50 LPRINT "This line is ELITE pitch."
60 LPRINT CHR$(27);"P";
70 LPRINT "This line is PICA pitch (normal)."
```

When you run this program you should get this:

```
This line is SEMI-CONDENSED pitch.
This line is ELITE pitch.
This line is PICA pitch (normal).
```

Line 20 turns on semi-condensed pitch with ⟨ESC⟩ "g". Line 30 prints the line at 15 characters per inch. Line 40 turns on elite pitch with ⟨ESC⟩ "M", and line 50 prints the line at 12 characters per inch. The ⟨ESC⟩ "P" in line 60 resets the printer to pica pitch and line 70 prints the line in pica pitch.

■ Expanded print

Each of the print pitches can be enlarged to twice its normal width. This is called expanded print. Try this program to see how it works:

```
10 ' Demo of expanded print
20 LPRINT "Demonstration of ";
30 LPRINT CHR$(14);
40 LPRINT "EXPANDED";
50 LPRINT CHR$(20);
60 LPRINT " printing."
70 LPRINT "Notice that ";
80 LPRINT CHR$(14);
90 LPRINT "EXPANDED mode"
100 LPRINT "automatically turns off at the end
    of a line."
```

```
Demonstration of EXPANDED printing.
Notice that EXPANDED mode
automatically turns off at the end of a line.
```

Expanded print set with CHR$(14) is automatically cancelled at the end of the line. This is convenient in many applications, such as for one line titles. Note that you didn't need to put an ⟨ESC⟩ in front of the CHR$(14), although ⟨ESC⟩ CHR$(14) works just the same.

You can also cancel one line expanded print *before* a carriage

return with CHR$(20), as done in line 50.

Sometimes you may wish to stay in expanded print for more than one line. Change your program to this:

```
10 ' Demo of permanent expanded mode
20 LPRINT CHR$(27);"W1";
30 LPRINT "Permanent expanded"
40 LPRINT "mode stays on until"
50 LPRINT "it is ";
60 LPRINT CHR$(27);"W0";
70 LPRINT "turned off."
```

Now the results look like this:

**Permanent expanded
mode stays on until
it is** turned off.

When you turn on expanded print with ⟨ESC⟩ "W" 1 it stays on until you turn it off with ⟨ESC⟩ "W" 0.

**Table 4-7
Expanded print commands**

Function	Control code
One line expanded ON	CHR$(14) *or* ⟨ESC⟩CHR$(14)
One line expanded OFF	CHR$(20)
Expanded ON	⟨ESC⟩ "W" 1
Expanded OFF	⟨ESC⟩ "W" 0

■ Condensed print

Each of the print pitches except the semi-condensed pitch also can be condensed to its normal width. This is called condensed print. Try this program to see how it works:

```
10 ' Demo of condensed print
20 LPRINT "Demonstration of ";
30 LPRINT CHR$(15);
40 LPRINT "CONDENSED";
50 LPRINT CHR$(18);
60 LPRINT " printing."
```

Demonstration of CONDENSED printing.

Condensed print set with CHR$(15) stays on until you turn it off with CHR$(18). Note that you don't need to put an ⟨ESC⟩ in front of the CHR$(15), although ⟨ESC⟩ CHR$(15) works just the same.

Table 4-8
Condensed print commands

Function	Control code
Condensed ON	CHR$(15) *or* ⟨ESC⟩ CHR$(15)
Condensed OFF	CHR$(18)

By combining expanded print and condensed print with the three pitches, this printer has ten different character widths available.

Enter this program to see how the print pitches, expanded print and condensed print can be combined:

```
10 ' Demo of various print pitches
20 LPRINT CHR$(15);
30 LPRINT CHR$(27);"M";
40 LPRINT "This line is CONDENSED ELITE pitch."
50 LPRINT CHR$(27);"P";
60 LPRINT "This line is CONDENSED PICA pitch."
70 LPRINT CHR$(18);
80 LPRINT CHR$(27);"g";
90 LPRINT "This line is SEMI-CONDENSED pitch."
100 LPRINT CHR$(27);"M";
110 LPRINT "This line is NORMAL ELITE pitch."
120 LPRINT CHR$(27);"P";
130 LPRINT "This line is NORMAL PICA pitch."
140 LPRINT CHR$(27);"W1";
150 LPRINT CHR$(15);
160 LPRINT CHR$(27);"M";
170 LPRINT "This line is EXPANDED CONDENSED ELITE."
180 LPRINT CHR$(27);"P";
190 LPRINT "This line is EXPANDED CONDENSED PICA."
200 LPRINT CHR$(18);
210 LPRINT CHR$(27);"g";
```

```
220 LPRINT "This is EXPANDED SEMI-CONDENSED."
230 LPRINT CHR$(27);"M";
240 LPRINT "This is EXPANDED ELITE."
250 LPRINT CHR$(27);"P";
260 LPRINT "This is EXPANDED PICA."
270 LPRINT CHR$(27);"W0"
280 END
```

Here's what you should get from this program:

This line is CONDENSED ELITE pitch.
This line is CONDENSED PICA pitch.
This line is SEMI-CONDENSED pitch.
This line is NORMAL ELITE pitch.
This line is NORMAL PICA pitch.
This line is EXPANDED CONDENSED ELITE.
This line is EXPANDED CONDENSED PICA.
This is EXPANDED SEMI-CONDENSED.
This is EXPANDED ELITE.
This is EXPANDED PICA.

■ Proportional printing

Have you ever looked closely at the printing in books and magazines? Doesn't it look nice? The main reason is that each character is given an amount of space proportional to its actual width. A typewriter (and most printers), on the other hand, give every character the same amount of space, no matter how wide it is. (Pica pitch, for example, gives a "w" and an "i" 1/10 of an inch each. Looking at these letters you see that a "w" is two or three times as wide as an "i".)

Well, you too can enjoy professional-looking proportional printing. You can turn proportional printing on and off with the following commands.

Table 4-9
Proportional commands

Function	Control code
Proportional ON	⟨ESC⟩ "p" 1
Proportional OFF	⟨ESC⟩ "p" 0

Try this program to see how the proportional spacing works.

```
10 ' Demo of proportional printing
20 LPRINT CHR$(27);"M";
30 LPRINT "This line is NORMAL ELITE printing."
40 LPRINT CHR$(27);"p1";
50 LPRINT "This line is PROPORTIONAL ELITE."
60 LPRINT CHR$(27);"P";
70 LPRINT "This line is PROPORTIONAL PICA."
80 LPRINT CHR$(27);"p0";
90 LPRINT "This line is NORMAL PICA printing."
100 END
```

When you run this program you should get this:

```
This line is NORMAL ELITE printing.
This line is PROPORTIONAL ELITE.
This line is PROPORTIONAL PICA.
This line is NORMAL PICA printing.
```

Line 20 selects the elite pitch and line 40 turns on the proportional printing with ⟨ESC⟩ "p" 1. Line 50 prints a line with proportional elite pitch. Then, line 60 selects the pica pitch, so that line 70 prints a line with proportional pica pitch. Finally, line 80 resets the proportional printing and line 90 prints a line in normal pica pitch.

NOTE: 1) You cannot use the proportional spacing with the draft characters.
2) When you set the "Print Pitch Panel" mode by the Print Pitch key on the control panel at power-on, these control codes related print pitches except expand print mode are ignored.

MAKING WORDS STAND OUT

Your printer has good print density when it's just printing normally. But sometimes you may want something to stand out from the rest of the page. This printer provides two ways to do

this: boldface and emphasized print. Both of these go over the characters twice, but they use slightly different methods to darken the characters. Let's try them and see what the difference is.

The following table shows the control codes for getting into and out of boldface and emphasized modes.

Table 4-10
Print emphasis commands

Function	Control code
Boldface ON	⟨ESC⟩ "G"
Boldface OFF	⟨ESC⟩ "H"
Emphasized ON	⟨ESC⟩ "E"
Emphasized OFF	⟨ESC⟩ "F"

Try them now with this little program:

```
10 ' Demo of boldface and emphasized
20 LPRINT CHR$(27);"G";
30 LPRINT "This line is BOLDFACE printing."
40 LPRINT CHR$(27);"E";
50 LPRINT "This line is BOLDFACE and EMPHASIZED."
60 LPRINT CHR$(27);"H";
70 LPRINT "This line is EMPHASIZED printing."
80 LPRINT CHR$(27);"F";
90 LPRINT "This line is normal printing."
100 END
```

Run this program. The results will look like this:

This line is BOLDFACE printing.
This line is BOLDFACE and EMPHASIZED.
This line is EMPHASIZED printing.
This line is normal printing.

Line 20 turns on boldface with ⟨ESC⟩ "G" and line 30 prints a line of text. In line 40 emphasized is turned on with ⟨ESC⟩ "E". Line 50 prints a line of text in boldface *and* emphasized. Line 60 then turns boldface off with ⟨ESC⟩ "H" so that line 70 can print in emphasized only. Finally, line 80 turns emphasized off, so

your printer is set for normal printing.

Look closely at the different lines of printing. In the line of boldface printing each character has been printed twice, and the paper was moved up slightly the second time they were printed. In emphasized printing, the characters are moved slightly to the right the second time the printer prints. The second line combined both of these so that each character was printed 4 times.

MIXING PRINT MODES

We have learned how to use the various print modes individually and together. Now we'll see how to combine them more efficiently.

You have at your disposal a unique command that lets you choose any valid combination of print modes and pitch. This is the Master Print mode command. It looks like this:

⟨ESC⟩ "!" n

Here, the value of n defines the print style to be selected. The value of n can range from 0 to 255, which is the range of values that can be stored in one eight-bit byte. If you look at each bit in this byte, you'll find that each one represents a printing style variation. Adding the binary values of the selected bits gives the value of n for a particular combination of print styles.

Table 4-11 shows the decimal values of the bits in the Master Print byte. To calculate the value n for a particular combination of printing styles, just add the decimal values of the features that you want to combine.

Table 4-11
Values of mixing print styles for Master Print

Bit	Print style	Decimal value
1	Elite print	1
2	Proportional print	2
3	Condensed print	4
4	Emphasized print	8
5	Boldface print	16
6	Expanded print	32
7	Italics	64
8	Underline	128

For example, if you want to select elite expanded boldface print, you would calculate the value of n like this:

Elite	1
Boldface	16
Expanded	32
$n=$	49

The command would look like this:

CHR$(27);"!";CHR$(49)

To better understand the way the print modes work, consider that each mode except pica (pica is the default) has a separate switch that can turned on and off via software. Once the switch is on, it stays on until turned off. However, when two modes that conflict are turned on at the same time, the printer must choose which one to use.

For example, suppose you turn on both Elite and Emphasized modes. These cannot combine, so the printer must make a choice; in this case, the printer chooses Elite.

Summary notes
1) Pica is the default pitch and is active when Elite is turned off.
2) When two modes conflict, the one of lesser priority bit-wise is cancelled. For example, Condensed and Emphasized can not be printed at the same time, printing is Emphasized.
3) Elite cancels Emphasized.
4) Underline, Expanded and Italics modes combine with any print modes.
5) Emphasized will not mix with Elite or Condensed.

CHAPTER 5

FORMATTING TEXT

Subjects covered in Chapter 5 include—
- **The carriage return and line feed**
- **The amount of space between lines**
- **Moving to the next page**
- **The number of lines on a printed page**
- **Horizontal and vertical tabs**
- **Setting margins — left, right, top and bottom**
- **Centering and aligning**

Chapter 4 showed us the basic techniques for using the printer. Now we're ready for the more advanced ones. We'll concentrate on changing the appearance of the page to suit our needs.

LINES AND LINE SPACING

■ Starting a new line

Up until now the only time we have thought about printing on a new line is when we *didn't* want it to happen. We learned that putting a semicolon (;) at the end of a BASIC line will *not* end the line of printing. So somehow, the computer tells the printer when to end one line and start another.

There are two codes that are used to end one line and start another. They are *carriage return* (CHR$(13)) and *line feed* (CHR$(10)). Like the escape codes, they have been given abbreviations which you'll find many texts (including this one): ⟨CR⟩ and ⟨LF⟩. The codes are simple, but their action is a little confusing (especially with BASIC). Let's begin with the carriage

return. Each time the printer receives a CHR$(13) it returns the print head to the left margin. It does not advance the paper (if DIP switch 2-6 is on; see below).

Line feed is more complicated. Each time the printer receives a CHR$(10) it both advances the paper one line and returns the print head to the left margin, ready to start a new line.

Now to add a little confusion — most (but not all) versions of BASIC add a line feed (CHR$(10)) to every carriage return (CHR$(13)) that they send. If your version of BASIC doesn't do this, then you should turn DIP switch 2-6 off, or send ⟨ESC⟩ "5" 1 command to the printer with the IBM-P mode, so that your printer will add the line feed for you. When you have DIP siwtch 2-6 off, or sent ⟨ESC⟩ "5" 1 command to the printer with the IBM-P mode, the printer will do the same thing when it receives a carriage return as it does when it receives a line feed.

If you find that your printer double spaces when it should single space, then you probably need to turn DIP switch 2-6 on, or send ⟨ESC⟩ "5" 0 command to the printer with the IBM-P mode.

■ Reverse line feeds

Your printer can move the paper up or down. The unique tractor design allows the paper to be fed in either direction without jamming. This allows you to move around the page at will. You can use this feature to print several columns of text side by side, or to print a graph and then move back up and insert descriptive legends. As you experiment you're bound to come up with more uses!

The simplest form of reverse paper feeding is a reverse line feed. The code is ⟨ESC⟩ ⟨LF⟩, which causes the paper to move down (in effect, moving the printing *up*) one line. A "line" used in a reverse line feed is the same size as a line in a regular line feed (this is normally 1/6 inch). When you change the line spacing (which you'll read about next), both forward and reverse line feeds change.

Table 5-1
Line feed commands

Function	Control code
Return print head to left margin	CHR$(13)
Advance paper one line	CHR$(10)
Reverse paper one line	⟨ESC⟩ CHR$(10)
Adds ⟨LF⟩ to every ⟨CR⟩	⟨ESC⟩ "5" 1 (IBM-P mode only)
Restore normal function of ⟨CR⟩	⟨ESC⟩ "5" 0 (IBM-P mode only)

■ Changing the line spacing

When you turn your printer on, the line spacing is set to 6 lines per inch (or 8 lines per inch if DIP switch 2-8 is off). This is fine for most printing applications, but when you want something different, the printer makes it easy to set the line spacing to whatever you want.

Try this program to see how easy it is to change the line spacing:

```
NEW
10 ' Demo of line spacing
20 FOR I=1 TO 25
30 IF I=13 THEN 60
40 LPRINT CHR$(27);"A";CHR$(I);
50 LPRINT "THIS LINE SPACING IS SET TO";I
60 NEXT I
70 LPRINT "LINE SPACING IS SET TO 1/6 INCH (NORMAL)."
80 LPRINT CHR$(27);"2"
90 END
```

Line 40 changes the line spacing. The command ⟨ESC⟩ "A" CHR$(n) changes the line spacing to n/60 of an inch. The loop that is started in line 20 increases the value of n (the variable I in this program) each time it is executed. So the line spacing increases as the program continues. Line 30 just shortcuts the loop when I = 13, since BASIC won't let us send CHR$(13) without adding an unwanted CHR$(10) to it. Finally, the ⟨ESC⟩ "2" in line 80 resets the line spacing to 6 lines per inch. This is a shortcut that is the same as ⟨ESC⟩ "A" CHR$(10).

This is what you will get with the Standard mode:

```
THIS LINE SPACING IS SET TO 4
THIS LINE SPACING IS SET TO 5
THIS LINE SPACING IS SET TO 6
THIS LINE SPACING IS SET TO 7
THIS LINE SPACING IS SET TO 8
THIS LINE SPACING IS SET TO 9
THIS LINE SPACING IS SET TO 10
THIS LINE SPACING IS SET TO 11
THIS LINE SPACING IS SET TO 12
THIS LINE SPACING IS SET TO 14
THIS LINE SPACING IS SET TO 15
THIS LINE SPACING IS SET TO 16
THIS LINE SPACING IS SET TO 17
THIS LINE SPACING IS SET TO 18
THIS LINE SPACING IS SET TO 19
THIS LINE SPACING IS SET TO 20
THIS LINE SPACING IS SET TO 21
THIS LINE SPACING IS SET TO 22
THIS LINE SPACING IS SET TO 23
THIS LINE SPACING IS SET TO 24
THIS LINE SPACING IS SET TO 25
LINE SPACING IS SET TO 1/6 INCH (NORMAL).
```

You may wonder why 1/60 of an inch was chosen as the increment for the line spacing command. There's a good reason: the dots that the printer makes are 1/180 inch apart. So this means that you can vary the line spacing in increments as fine as three dots — unless you want finer spacing, like one dot spacing.

The ⟨ESC⟩ "3" CHR$(*n*) command sets the line spacing in increments of 1/180 inch. Change line 40 in your program to this:

```
40 LPRINT CHR$(27);"3";CHR$(I);
```

and run the program again. Now the results will look like this:

```
THIS LINE SPACING IS SET TO 9
THIS LINE SPACING IS SET TO 10
THIS LINE SPACING IS SET TO 14
THIS LINE SPACING IS SET TO 15
THIS LINE SPACING IS SET TO 16
THIS LINE SPACING IS SET TO 17
THIS LINE SPACING IS SET TO 18
THIS LINE SPACING IS SET TO 19
THIS LINE SPACING IS SET TO 20
THIS LINE SPACING IS SET TO 21
THIS LINE SPACING IS SET TO 22
THIS LINE SPACING IS SET TO 23
THIS LINE SPACING IS SET TO 24
THIS LINE SPACING IS SET TO 25
LINE SPACING IS SET TO 1/6 INCH (NORMAL).
```

The program works just the same as before, but the line spacings are just one-third what they were. This is because ⟨ESC⟩ "3" CHR$(*n*) sets the line spacing to *n*/180 inch.

Now, let's consider the IBM modes. When you run the first program with the IBM modes, you cannot change the line spacing. The ⟨ESC⟩ "A" CHR$(*n*) command in the IBM modes only defines the line spacing; the ⟨ESC⟩ "2" command changes the line spacing to the amount defined by the previous ⟨ESC⟩ "A"

For the IBM modes, you need to change the following lines in the first program as shown below:

```
40 LPRINT CHR$(27);"A";CHR$(I);CHR$(27);"2";
80 LPRINT CHR$(27);"A";CHR$(12);CHR$(27);"2"
```

and run the program again. Now the results will look like this:

```
THIS LINE SPACING IS SET TO 2
THIS LINE SPACING IS SET TO 5
THIS LINE SPACING IS SET TO 6
THIS LINE SPACING IS SET TO 8
THIS LINE SPACING IS SET TO 9
THIS LINE SPACING IS SET TO 10
THIS LINE SPACING IS SET TO 11
THIS LINE SPACING IS SET TO 12
THIS LINE SPACING IS SET TO 14
THIS LINE SPACING IS SET TO 15
THIS LINE SPACING IS SET TO 16
THIS LINE SPACING IS SET TO 17
THIS LINE SPACING IS SET TO 18
THIS LINE SPACING IS SET TO 19
THIS LINE SPACING IS SET TO 20
THIS LINE SPACING IS SET TO 21
THIS LINE SPACING IS SET TO 22
THIS LINE SPACING IS SET TO 23
THIS LINE SPACING IS SET TO 24
THIS LINE SPACING IS SET TO 25
LINE SPACING IS SET TO 1/6 INCH (NORMAL).
```

By comparing with the results between the Standard mode and the IBM modes, you'll notice that the ⟨ESC⟩ "A" CHR$($n$) command with the IBM modes defined the line spacing as $n/72$ of an inch, instead of $n/60$ of an inch.

This is because that the IBM Graphics printer and Proprinter are normally sets the line spacing to $n/72$ of an inch with this command, and most software programs use this command to set the line spacing. And we've thought the compatibility with those programs. In addition, the ⟨ESC⟩ "3" CHR$($n$) command sets the line spacing to $n/216$ of an inch with the same reason.

Table 5-2 shows all the line spacing commands, including several "shortcut" commands for commonly used line spacings.

Table 5-2
Line spacing commands

Function	Control code
Set line spacing to 1/8 inch	⟨ESC⟩ "0"
Set line spacing to 7/60 inch	⟨ESC⟩ "1" (Standard mode only)
Set line spacing to 7/72 inch	⟨ESC⟩ "1" (IBM modes only)
Set line spacing to 1/6 inch	⟨ESC⟩ "2" (Standard mode)
	⟨FS⟩ "2" (IBM modes)
Set line spacing to n/60 inch	⟨ESC⟩ "A" CHR$(n) (Standard mode only)
Set line spacing to n/72 inch	⟨FS⟩ "A" CHR$(n) (IBM modes only)
Set line spacing to n/180 inch	⟨ESC⟩ "3" CHR$(n) (Standard modes only)
Set line spacing to n/216 inch	⟨ESC⟩ "3" CHR$(n) (IBM modes only)
Define line spacing to n/72 inch	⟨ESC⟩ "A" CHR$(n) (IBM modes only)
Use ⟨ESC⟩ "A" definition	⟨ESC⟩ "2" (IBM modes only)
One-time line feed of n/180 inch	⟨ESC⟩ "J" CHR$(n) (Standard mode only)
One-time line feed of n/216 inch	⟨ESC⟩ "J" CHR$(n) (IBM modes only)
One-time reverse line feed of n/180 inch	⟨ESC⟩ "j" CHR$(n) (Standard mode only)
One-time reverse line feed of n/216 inch	⟨ESC⟩ "j" CHR$(n) (IBM modes only)

■ **Moving down the page without a carriage return**

So far, all the commands that move the paper also move the print head back to the left margin. Normally this is what you want. Sometimes, though, you may wish to move down the page without moving the printhead back to the left margin. The following commands do just that.

The ⟨ESC⟩ "J" CHR$(n) command causes the printer to make one line feed of $n/180$ inch ($n/216$ inch with the IBM modes), but does not *change* the setting of the line spacing. Try this program to see how it works:

```
NEW
10 ' Demo of one-time line feeds
20 LPRINT "LINE NUMBER 1."
30 LPRINT "LINE NUMBER 2.";
40 ' One-time line feed
50 LPRINT CHR$(27);"J";CHR$(100);
60 LPRINT "LINE NUMBER 3."
70 LPRINT "LINE NUMBER 4."
80 END
```

Here is what your printer will produce with Standard mode:

```
LINE NUMBER 1.
LINE NUMBER 2.


              LINE NUMBER 3.
LINE NUMBER 4.
```

The ⟨ESC⟩ "J" CHR$(100) in line 50 changes the spacing to 100/180 inches for one line only without moving the printhead. The rest of the lines printed with the normal line spacing. Notice that both line 30 and line 50 end with semicolons. This prevents the normal line feed from occurring.

The ⟨ESC⟩ "j" CHR$(*n*) command works the same way except that the paper moves in the opposite direction. Try this simple change to your program to see the difference.

```
40 ' One-time reverse line feed
50 LPRINT CHR$(27);"j";CHR$(100);
```

```
              LINE NUMBER 3.
LINE NUMBER 4.
LINE NUMBER 1.
LINE NUMBER 2.
```

PAGE CONTROL

Now that we have seen how to control line spacing, we can go on to page control — positioning the printing on the page and adjusting the paper length.

■ Form feed

The simplest forms control code is the *form feed*. Form feed (or ⟨FF⟩) is CHR$(12) and causes the printer to move the paper to the top of the next sheet. Try it by changing lines 40 and 50 to this:

```
40 ' Form feed
50 LPRINT CHR$(12);
```

Before you run the program, turn your printer off and adjust the paper so that the top of the sheet is even with the top of the ribbon guide on the print head, then turn the printer back on. If you don't remember how to do this, review Chapter 2. When you run the program, the results will look like this:

```
       ┌───┬─────────────────────
       │   │ LINE  NUMBER  1.
   ◯   │   LINE  NUMBER  2.
       │
       │              ┌───┬────────────────────
   ◯   │              │   │ LINE  NUMBER  3.
       │          ◯   │   LINE  NUMBER  4.
       │              │
```
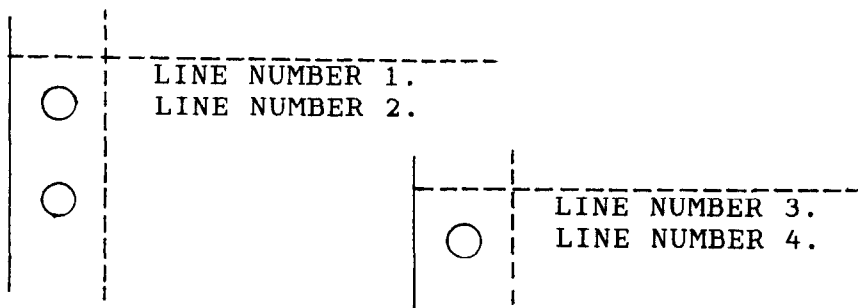
The form feed (CHR$(12)) in line 50 caused the printer to move to the top of a new page before printing the last two lines.

■ Reverse form feed

Just as your printer can perform a reverse line feed, it can do a reverse form feed. This code moves the paper so that the print head is positioned at the top of the current page. This can be used, for example, to print text in a multi-column magazine format; print the first column, then reverse form feed back to the top of the page to start the second column. The code for reverse form feed is easy to remember: ⟨ESC⟩ ⟨FF⟩.

Table 5-3
Form feed commands

Function	Control code
Advance paper to top of next page	CHR$(12)
Reverse paper to top of current page	⟨ESC⟩ CHR$(12)
Set the top of form to the current position	⟨ESC⟩ "4" (IBM-P mode only)

■ Setting the top of form

When you turn on the printer, the top of form is automatically set to the current position. If you want to change the position, you can re-set it by the following control code with the IBM-P mode.

⟨ESC⟩ "4"

In addition, you can re-set it with the control panel as described in Chapter 3."

■ Changing the page length

You may have some computer forms that are not 11 inches high. That's no problem, because you can tell your printer how high the forms are that you are using. There are two commands for doing this, shown in this table.

Table 5-4
Form length control

Function	Control code
Set the page length to n lines	⟨ESC⟩ "C" CHR$(n)
Set the page length to n inches	⟨ESC⟩ "C" CHR$(0) CHR$(n)

Let's set up a 7 inch high form length, which is typical of many computer checks. The following program will do it.

```
NEW
10 ' Demo of variable form lengths
20 LPRINT CHR$(27);"C";CHR$(0);CHR$(7);
30 LPRINT "PAY TO THE ORDER OF:"
40 LPRINT CHR$(12);
50 LPRINT "PAY TO THE ORDER OF:"
60 END
```

This program should print "PAY TO THE ORDER OF:" twice, and they should be 7 inches apart. Line 20 sets the form length to 7 inches. After line 30 prints, line 40 sends a form feed advance the paper to the top of the next form. Line 50 then prints its message.

After you have run this program, turn off the printer and adjust the top of form position. When you turn the printer back on the page length will reset to its normal setting (usually 11 inches).

TOP AND BOTTOM MARGINS

Some programs don't keep track of where they are printing on the page. This causes problems when you get to the bottom of a page because these programs just keep on printing, right over the perforation. This makes it very hard to read, especially if a line happens to fall right on the perforation.

The printer has a solution to this predicament. This printer can keep track of the position on the page, and advance the paper so that you won't print too near the perforation. There are two commands to do this. One controls the space at the top of the page (top margin) and the other controls the space at the bottom of the page (bottom margin). The control codes are given in the following table:

Table 5-5
Top and bottom margin commands

Function	Control code
Set top margin to *n* lines	⟨ESC⟩ "r" CHR$(*n*)
Set bottom margin to *n* lines	⟨ESC⟩ "N" CHR$(*n*)
Cancel top and bottom margins	⟨ESC⟩ "O"

In both cases the value of *n* tells your printer how many lines to skip, although there is a slight difference in the usage. When you set the top margin with ⟨ESC⟩ "r" CHR$(*n*), the value of *n* tells the printer what line to start printing on. When you set the bottom margin with ⟨ESC⟩ "N" CHR$(*n*), the value of *n* tells the printer how many blank lines should be left at the bottom of the page.

Let's try a simple application to see how these margins work. Enter this program, which will print 150 lines *without* top and bottom margins.

```
10 ' Demo of top and bottom margins
60 LPRINT CHR$(12);   : ' form feed
70 FOR I=1 TO 150
80 LPRINT "THIS IS LINE";I
90 NEXT I
110 LPRINT CHR$(12);   : ' form feed
120 END
```

When you run this program it will print 150 lines right down the page and across the perforations. When it's done line 110 sends a form feed to advance to the top of the next page.

Now add the following lines to your program. (Don't forget the semicolons or you won't get quite the same results that we did.)

```
20 ' Leave 6 blank lines at the bottom of page
30 LPRINT CHR$(27);"N";CHR$(6);
40 ' Start top of page at line 6
50 LPRINT CHR$(27);"r";CHR$(6);
100 LPRINT CHR$(27);"O"   : ' clear top and
    bottom margins"
```

Now when you run the program, your printer skip the first six lines and the last six lines on each page. Always send a form feed after setting the top margin, or it will not work on the first page printed. That's because the top margin only takes effect after a form feed.

Line 50 sets the top margin, line 30 sets the bottom margin, and line 100 clears both margins when we are done.

```
THIS IS LINE 1
THIS IS LINE 2
THIS IS LINE 3
THIS IS LINE 4
THIS IS LINE 5
THIS IS LINE 6
THIS IS LINE 7
THIS IS LINE 8
THIS IS LINE 9
THIS IS LINE 10
```

```
THIS IS LINE 49
THIS IS LINE 50
THIS IS LINE 51
THIS IS LINE 52
THIS IS LINE 53
THIS IS LINE 54
THIS IS LINE 55
```

```
THIS IS LINE 56
THIS IS LINE 57
THIS IS LINE 58
THIS IS LINE 59
THIS IS LINE 60
THIS IS LINE 61
THIS IS LINE 62
THIS IS LINE 63
THIS IS LINE 64
```

```
THIS IS LINE 103
THIS IS LINE 104
THIS IS LINE 105
THIS IS LINE 106
THIS IS LINE 107
THIS IS LINE 108
THIS IS LINE 109
THIS IS LINE 110
```

```
THIS IS LINE 111
THIS IS LINE 112
THIS IS LINE 113
THIS IS LINE 114
THIS IS LINE 115
THIS IS LINE 116
THIS IS LINE 117
THIS IS LINE 118
THIS IS LINE 119
THIS IS LINE 120
THIS IS LINE 121
```

SETTING LEFT AND RIGHT MARGINS

The left and right margins of this printer work just like a typewriter — once they are set all printing is done between them. You can set margins either by using the keys on the control panel, as described in Chapter 3, or by escape commands. The commands to set the margins are given in the following table:

Table 5-6
Left and right margin commands

Function	Control code
Set left margin at column *n*	⟨ESC⟩ "l" CHR$(*n*)
Set right margin at column *n*	⟨ESC⟩ "Q" CHR$(*n*) (except IBM-P mode)
	⟨FS⟩ "Q" CHR$(*n*)(IBM-P mode)
Set left margin at column *n1* and right margin at column *n2*	⟨ESC⟩ "X" CHR$(*n1*) CHR$(*n2*)

Try setting the margins with this program:

```
10 ' Demo of margins
20 GOSUB 70
30 LPRINT CHR$(27);"l";CHR$(10);
40 LPRINT CHR$(27);"Q";CHR$(70);
50 GOSUB 70
60 END
70 FOR I=1 TO 80
80 LPRINT "X";
90 NEXT I
100 LPRINT
110 RETURN
```

The first thing that this program does is to branch to the subroutine that starts in line 70. This subroutine prints 80 X's in a row. The first time that the subroutine is used, all the X's fit in one line. Then line 30 sets the left margin to 10, and line 40 sets the right margin to 70. Once again the subroutine is used, but this time the X's won't all fit on one line since there is now only room for 60 characters between the margins.

Run the program. The results will look like this:

```
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
     xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
     xxxxxxxxxxxxxxxxxxxx
```

When you want to reset the margins to the default values, you have two choices. You can either turn the printer off and back on, or you can set margin values equal to the default values. This means that you should set a left margin of 0 and right margin of 80 or 136 in pica pitch.

If you change the pitch of your printing after you set your margins, the margins will not change. They stay at the same place on the page. So if you set the margins to give you 65 columns of printing when you are using pica type, and you change to elite type, you will have room for more than 65 columns of elite printing between the margins.

HORIZONTAL AND VERTICAL TABS

Suppose you need to move across the page to a certain position several times in a document. It's not much fun to type in space after space. You don't have to — you can "tab" your way across the page.

Your printer's tabs are like those on a typewriter, but much more powerful. You have both horizontal and vertical tabs which can be used for both text and graphics — and they're really handy for indenting paragraphs and making tables.

■ Horizontal tabs
Horizontal tabs are set automatically every eight positions. To move the print head to the next tab position, send CHR$(9).
Try this program to see how the default tabs work.

```
10 ' Demo of horizontal tabbing
20 LPRINT "ONE";CHR$(9);"TWO";CHR$(9);"THREE";
   CHR$(9);"FOUR"
```

Here's what you should get—

ONE   TWO   THREE   FOUR

Even though the words are different length, they are spaced out evenly by the horizontal tabs.

Now add the following line to your program to set different horizontal tabs:

```
15 LPRINT CHR$(27);"D";CHR$(7);CHR$(14);CHR$(21);
   CHR$(0);
```

⟨ESC⟩ "D" is the command to begin setting horizontal tabs. It must be followed by characters representing the positions where you want the tabs set. In our program we are setting tabs in columns 7, 14, and 21. The final CHR$(0) ends the string of tabs. In fact, any character that is not greater than the provious one will stop setting tabs. This means that you must put all your tab values in order, from least to greatest, or they won't all get set.

When you run the program now it produces this:

ONE   TWO   THREE   FOUR

The words are now closer together, but still evenly spaced. To reset the default tabs, use one of the following programs:

```
LPRINT CHR$(27);"e0";CHR$(8)
```

```
LPRINT CHR$(27);"R"
```

The ⟨ESC⟩ "e" 0 CHR$(n) command sets the horizontal tabs to every n columns, and the ⟨ESC⟩ "R" command restores the default tab positions.

Table 5-7
Horizontal tab commands

Function	Control code
Advance to next tab position	CHR$(9)
Set tabs at *n1*, *n2*, etc.	⟨ESC⟩ "D" CHR$(*n1*) CHR$(*n2*) ..... CHR$(0)
Set tabs every *n* columns	⟨ESC⟩ "e" 0 CHR$(*n*)
Restore the default tab positions	⟨ESC⟩ "R" (IBM-P mode only)

■ One-time horizontal tabs

Suppose you need to move to a position across the page, but you only need to do it once. It doesn't make much sense to set up a tab to use only one time. There is an easier way.

The solution is called a *one-time* tab. Table 5-8 show the three commands.

Table 5-8
One-time horzontal tabs

Function	Control code
One-time horizontal tab of *n* spaces	⟨ESC⟩ "f" 0 CHR$(*n*)
Absolute horizontal tab	⟨ESC⟩ "$" CHR$(*n1*) CHR$(*n2*)
Relative horizontal tab	⟨ESC⟩ "\" CHR$(*n1*) CHR$(*n2*) (except IBM-P mode)
	⟨FS⟩ "\" CHR$(*n1*) CHR$(*n2*) (IBM-P mode)

The first command in Table 5-8 moves the print head *n* columns to the right. It has the same effect as sending *n* spaces to the printer.

The absolute horizontal tab command moves the print head to a specified position on the page. The position that you want the print head to move to (measured from the left margin) is specified by the values of *n1* and *n2* in 1/60-inch units using the formula $n1 + n2 \times 256$.

Try this program to see how this works.

```
10 ' Demo of absolute horizontal tabs
20 FOR I=60 TO 70 STEP 2
30 LPRINT I;"+++++";
```

```
40 LPRINT CHR$(27);"$";CHR$(I);CHR$(0);
50 LPRINT "Horizontal tab."
60 NEXT I
70 END
```

In this program, the print head is positioned before the "H" in "Horizontal" is printed.

```
60 +++++ Horizontal tab.
62 +++++ Horizontal tab.
64 +++++ Horizontal tab.
66 +++++ Horizontal tab.
68 +++++ Horizontal tab.
70 +++++ Horizontal tab.
```

The relative horizontal tab command can move the print head right from the current position. The formula for calculating how much the print head moves is the same as in the absolute horizontal tab command. However, the units by which the print head actually moves vary with the print pitch you are using, as shown in Table 5-9.

Table 5-9
Units of motion for the relative
horizontal tab commands

Print pitch	Unit of motion	
	Draft characters	Letter Quality characters
Pica pitch	1/120 inch	1/180 inch
Elite pitch	1/120 inch	1/180 inch
Semi-condensed & Condensed pitch	1/240 inch	1/180 inch

Try this program to see how this works.

```
10 ' Demo of relative horizontal tabs
20 FOR I=1 TO 3
30 LPRINT "Relative";
```

```
40 LPRINT CHR$(27);"\";CHR$(I*20);CHR$(0);
50 LPRINT "Horizontal";
60 LPRINT CHR$(27);"\";CHR$(I*10);CHR$(0);
70 LPRINT "Tab."
80 NEXT I
90 END
```

```
Relative  Horizontal Tab.
Relative   Horizontal  Tab.
Relative    Horizontal   Tab.
```

■ **Vertical tabs**

Vertical tabs have the same kinds of uses that horizontal tabs do — they just work in the other direction. Horizontal tabs allow you to reach a specific column on the page no matter where you start from. Vertical tabs are the same. If you have a vertical tab set at line 20, a *vertical tab* (or ⟨VT⟩) will move you to line 20 whether you start from line 5 or line 19.

Vertical tabs are *not* set at the power-on default. If you send a CHR$(11) which is the ASCII code for ⟨VT⟩ before we have set up tabs, the command advances the paper only one line. Enter this program to see how this works.

```
10 ' Demo of vertical tabs
40 LPRINT CHR$(11);"FIRST TAB."
50 LPRINT CHR$(11);"SECOND TAB."
60 LPRINT CHR$(11);"THIRD TAB."
70 LPRINT CHR$(11);"FOURTH TAB."
```

Now, let's set some vertical tabs of our own. Add these lines to the program:

```
20 LPRINT CHR$(27);"B";CHR$(10);CHR$(15);
30 LPRINT CHR$(25);CHR$(30);CHR$(0);
```

⟨ESC⟩ "B" is the command to set vertical tabs. Like the horizontal tab setting command, tab positions must be defined

in ascending order. Our example sets vertical tabs at lines 10, 15, 25 and 30. Then the CHR$(11) in each of the following lines advances the paper to the next vertical tab. The printout is shown below.

```
FIRST TAB.
```

```
SECOND TAB.
```

```
THIRD TAB.
```

```
FOURTH TAB.
```

Add one more line to the program to demonstrate one more feature of vertical tabs.

```
80 LPRINT CHR$(11);"FIFTHH TAB."
```

Now when you run the program the first page looks just like before, but line 80 sends one more ⟨VT⟩ than there are tabs. This doesn't confuse the printer — it advances the paper to the *next* tab position, which happens to be the first tab position on the next page.

There's a one-time vertical tab command that works just like

the one-time horizontal tab command. Try this program to see how these commands work.

```
10 ' Demo of print positions
20 LPRINT "* 1st Line, 1st Column"
30 LPRINT CHR$(27);"f1";CHR$(3);
40 LPRINT "* 5th Line, 1st Column"
50 LPRINT CHR$(27);"f0";CHR$(7);
60 LPRINT "* 6th Line, 7th Column"
70 LPRINT CHR$(27);"f1";CHR$(2);
80 LPRINT CHR$(27);"f0";CHR$(5);
90 LPRINT "* 9th Line, 5th Column"
100 END
```

In this program, line 30 sends a one-time vertical tab with ⟨ESC⟩ "f" 1 command. Line 50 sends a one-time horizontal tab with ⟨ESC⟩ "f" 0 command. Finally, line 70 and line 80 send one-time vertical tab and one-time horizontal tab, so the line 90 prints text at the specified position.

```
* 1st Line, 1st Column.


* 5th Line, 1st Column.
      * 6th Line, 7th Column.


   * 9th Line, 5th Column.
```

Table 5-10
Vertical tab commands

Function	Control code
Advance paper to next tab position	CHR$(11)
Set vertical tabs at *n1*, *n2*, etc.	⟨ESC⟩ "B" CHR$(*n1*) CHR$(*n2*) ..... CHR$(0)
Set vertical tabs every *n* lines	⟨ESC⟩ "e" 1 CHR$(*n*)
One-time vertical tab	⟨ESC⟩ "f" 1 CHR$(*n*)

■ Vertical tab channels

Vertical tab channels are especially hefpful in two situations. The first occurs when you are writing a program to accompany a preprinted form that can accommodate verious types of responses. The second occurs when you create a multipage form or report with different vertical tabs on each page.

Table 5-11
Vertical tab channel commands

Function	Control code
Set vertical tabs at $n1$, $n2$, etc. as channel $n0$	$\langle$ESC$\rangle$ "b" CHR\$($n0$) CHR\$($n1$) CHR\$($n2$) ..... CHR\$(0)
Select vertical channel $n0$	$\langle$ESC$\rangle$ "/" CHR\$($n0$)

You can store up to eight channels of tab stops. They are numbered from 0 to 7. If you have already stored a set using $\langle$ESC$\rangle$ "B" command, your printer has labelled it as channel 0.

Try this program how to see the vertical tab channels work.

```
10 ' Demo of vertical tab channels
20 LPRINT CHR$(27);"b";CHR$(1);CHR$(10);CHR$(20);
  CHR$(0);
30 LPRINT CHR$(27);"b";CHR$(2);CHR$(15);CHR$(25);
  CHR$(0);
40 LPRINT CHR$(27);"b";CHR$(3);CHR$(17);CHR$(28);
  CHR$(0);
50 ' Use vertical tab channels
60 FOR I=1 TO 3
70 LPRINT "TOP OF FORM"
80 LPRINT CHR$(27);"/";CHR$(I);
90 LPRINT CHR$(11);
100 LPRINT "1ST TAB OF CHANNEL";I
110 LPRINT CHR$(11);
120 LPRINT "2ND TAB OF CHANNEL";I
130 LPRINT CHR$(12);
140 NEXT I
150 LPRINT CHR$(27);"@"
160 END
```

When you run this program the printout should look like this.

```
┌ TOP OF FORM ──────────┐  ┌ TOP OF FORM ──────────┐  ┌ TOP OF FORM ──────────┐
│                       │  │                       │  │                       │
│                       │  │                       │  │                       │
│                       │  │                       │  │                       │
│                       │  │                       │  │                       │
│ 1ST  TAB  OF  CHANNEL  1 │                       │  │                       │
│                       │  │                       │  │                       │
│                       │  │ 1ST  TAB  OF  CHANNEL  2 │                       │
│                       │  │                       │  │ 1ST  TAB  OF  CHANNEL  3 │
│                       │  │                       │  │                       │
│ 2ND  TAB  OF  CHANNEL  1 │                       │  │                       │
│                       │  │                       │  │                       │
│                       │  │ 2ND  TAB  OF  CHANNEL  2 │                       │
│                       │  │                       │  │                       │
│                       │  │                       │  │ 2ND  TAB  OF  CHANNEL  3 │
│                       │  │                       │  │                       │
│                       │  │                       │  │                       │
└                       └  └                       └  └                       └
```

In this program we set tabs at 10 and 20 in channel 1 in line 20. In line 30 we set tabs 15 and 25 in channel 2, and in line 40 we set tabs 17 and 28 in channel 3.

Because the channels are stored, you must make the printer to recall one before you use it, so we used ⟨ESC⟩ "/" CHR$($n0$) in line 80.

CENTERING AND ALIGNING TEXT

Text can be arranged in any of four formats: left aligned or left-justified (normal printing with "ragged" right margin), centered between the margins, right aligned (right-justified) or fully aligned. These are selected by the following commands:

Table 5-12
Alignment commands

Function	Control code
Left-justified printing	⟨ESC⟩ "a" 0
Centered printing	⟨ESC⟩ "a" 1
Right-justified printing	⟨ESC⟩ "a" 2
Fully-justified printing	⟨ESC⟩ "a" 3

Try this program to see how easy it is.

```
10 ' Demo of aligning and centering
20 LPRINT CHR$(27);"1";CHR$(20);
30 LPRINT CHR$(27);"Q";CHR$(60);
40 LPRINT CHR$(27);"a0";
50 LPRINT "THIS LINE IS LEFT JUSTIFIED."
6C LPRINT CHR$(27);"a1";
70 LPRINT "THIS LINE IS CENTERED."
80 LPRINT CHR$(27);"a2";
90 LPRINT "THIS LINE IS RIGHT JUSTIFIED."
100 LPRINT CHR$(27);"a3";
110 LPRINT "THIS LINE IS LEFT AND RIGHT (FULL) JUSTIFIED."
```

When you run this program, you should get like this:

```
THIS LINE IS LEFT JUSTIFIED.
        THIS LINE IS CENTERED.
            THIS LINE IS RIGHT JUSTIFIED.
THIS   LINE   IS   LEFT   AND   RIGHT   (FULL)
JUSTIFIED.
```

CHAPTER 6

SPECIAL FEATURES OF THE PRINTER

Subjects covered in Chapter 6 include—
- **Printer's bell**
- **Master reset**
- **Uni-directional printing**
- **International character sets**
- **Printing BIG characters**
- **The optional sheet feeder**
- **Reading a hex dump**

In the previous chapters we have learned about several groups of control codes. In this chapter we will look at more control codes. These codes don't fit neatly into any of the groupings that we have looked at so far, but they add a lot of useful features to your printer.

■ **Now hear this**

You may have heard the printer's *bell* if you have ever run out of paper. And you may have wondered why it's called a bell when it *beeps* instead of ringing! It's a long story that goes back to the early days of computers, when teletype machines were used for computer terminals. These mechanical marvels were equipped with a bell that could be heard for blocks. This bell was used to signal the operator that something needed attention. The code that the computer sent to the teletype machine to ring the bell was, reasonably enough, called a *bell code*. The name *bell code* is still with us, even if the bell has changed to a beeper, and a lot of people still call the beeper a bell, even if it doesn't sound like one.

The code to sound the "bell" is CHR$(7), which is ASCII code 7 or ⟨BEL⟩. Any time your printer receives this code it will sound the bell for a quarter of a second. This can be used to remind the operator to change the paper or to make other adjustments to the printer.

You can try this by typing:

```
LPRINT CHR$(7)
```

■ Resetting the printer

Up to now when we wanted to reset the printer to the power on condition we have had to either turn the printer off and on again, or to send the specific codes that reset the particular features. There is an easier way. The control code ⟨ESC⟩ "@" will reset all of the printer's features to the power on condition (as determined by the DIP switches), with two exceptions. The exceptions are that ⟨ESC⟩ "@" will not erase any characters that you have stored in the printer's RAM memory (Chapter 7 tells you how to create your own characters), and if you set the "Panel" modes by the control panel settings at power on, these functions will remain with this control code.

■ Taking the printer off line

You know how to put your printer *off line* with the On Line key on the control panel. The printer has another *off line* state that can be controlled from the computer. When you turn the printer *off line* from the computer, the printer will ignore anything that you send it, except for the code to go *on line* again. CHR$(19) (⟨ESC⟩ "Q" 3 with the IBM-P mode) is the code to turn the printer off line; CHR$(17) turns the printer back to on line status.

■ Printing the bottom of the sheet

Sometimes when you are using individual sheets of paper you may want to print near the bottom of a sheet. The paper-out detector usually stops the printer when you are about 1 inch from the bottom of the sheet. This is notify you if you are running out of continuous paper.

Your printer has the ability to print right to the bottom of the

sheet. You can disable the paper-out detector so that it doesn't stop the printer. This will allow you to print to the end of the sheet, and even beyond if you are not careful. The codes to control the paper-out detector, along with the other codes that we have just looked at are in the following table.

Table 6-1
Some miscellaneous commands

Function	Control code
Sound bell	CHR$(7)
Master rest	⟨ESC⟩ "@"
Off line	CHR$(19) (except IBM-P mode)
	⟨ESC⟩ "Q" 3 (IBM-P mode)
On line	CHR$(17)
Paper-out detector off	⟨ESC⟩ "8"
Paper-out detector on	⟨ESC⟩ "9"
Move print head back one space	CHR$(8)
Delete last character sent	CHR$(127)
Cancel text in print buffer	CHR$(24)
Print "slash zero"	⟨ESC⟩ "~" 1
Print "normal zero"	⟨ESC⟩ "~" 0
Immediate-print on	⟨ESC⟩ "i" 1
Immediate-print off	⟨ESC⟩ "i" 0
Add *n* dot spaces between characters	⟨ESC⟩ CHR$(32) CHR$(*n*) (Standard mode only)
Half-speed print on	⟨ESC⟩ "s" 1
Half-speed print off	⟨ESC⟩ "s" 0

■ Backspace, delete, and cancel text

Backspace (CHR$(8)) "backs up" the print head so that you can print two characters right on top of each other. Each time your printer receives a backspace it moves the print head one character to the left, instead of to the right. You can *strike over* multiple letters by sending more than one backspace code.

Delete (CHR$(127)) also "backs up" one character, but then it "erases" the previous character (it's erased from your printer's buffer, not from the paper).

Cancel text (CHR$(24)) deletes all the text in the print buffer; that is, in the line before the delete text command. Since your printer prints one line of text at a time, only that line will be deleted.

The following program shows how these codes works.

```
10 LPRINT "BACKSPACE DOES NOT";
20 LPRINT CHR$(8);CHR$(8);CHR$(8);
30 LPRINT "=== WORK"
40 LPRINT "DELETE DOES NOT";
50 LPRINT CHR$(127);CHR$(127);CHR$(127);
60 LPRINT "WORK"
70 LPRINT "CANCEL LINE";
80 LPRINT CHR$(24);
90 LPRINT "DOES NOT WORK"
```

Here is what this program will print:

```
BACKSPACE DOES N̶O̶T̶ WORK
DELETE DOES WORK
DOES NOT WORK
```

The backspace codes in line 20 move the print head a total of three spaces to the left so that the first part of line 30 will over-print the word "NOT". The delete codes in line 50 "erase" the three letters in the word "NOT" so that it doesn't even print.

In line 80, CHR$(24) deletes the words in line 70. The semicolon at the end of line 70 prevents a line feed from causing that line to print before the printer receives the CHR$(24) code. The text in line 90 prints as it normally would because it is after CHR$(24).

■ Printing zeroes
There are two types of zeroes. There is the one we use every day — 0 — and this is what your printer will print as its default.

The other type is used almost exclusively in computers and engineering. It is called the "slash zero" and is written like this — ∅. The line through the number prevents you from misreading it as the letter "O".

■ Immediate-print
This printer can print at the quick rate of 180 characters per second with draft characters. But it will also print more slowly

at the speed of your typing. In the *immediate-print* mode, the print head prints one character at a time, as you send it. The printer also moves the paper up so that you can see the current line and then down to continue printing.

You can turn the immediate-print mode on with ⟨ESC⟩ "i" 1. Before looking at it, let's review the normal operation of the print buffer. Enter this program.

```
20 A$="" : INPUT " TYPE A CHARACTERS ",A$
30 IF A$="" THEN 50
40 LPRINT A$; : GOTO 20
50 LPRINT :LPRINT CHR$(27);"@"
```

Now type several characters, and after each press the RETURN key. The printer simply puts the characters into its buffer while it waits for a carriage return code. (In this program the RETURN key doesn't send a carriage return code.) To end this program and print the contents of the buffer, press RETURN alone.

Now add this line:

```
10 LPRINT CHR$(27);"i1";
```

And RUN the program. Your printer will now respond to your typing — immediately.

When you are finished, press RETURN alone.

■ Adjusting the width of space between characters

This printer provides a command that adjusts the space between all the characters that it prints with the Standard mode. The ⟨ESC⟩ CHR$(32) CHR$(n) command adds blank space between the characters. The units of space that are added vary with the print pitch you are using, as shown in Table 6-2.

Table 6-2
Units of motion for the adjustment
of the character width

Print pitch	Unit of motion	
	Draft characters	Letter Quality characters
Pica pitch	1/120 inch	1/180 inch
Elite pitch	1/120 inch	1/180 inch
Semi-condensed & Condensed pitch	1/240 inch	1/360 inch

Try this program to see how this works.

```
10 ' Demo of adjusting spaces
20 LPRINT CHR$(27);"x1";
30 FOR I=10 TO 1 STEP -2
40 LPRINT CHR$(27);" ";CHR$(I);
50 LPRINT "This line is added";I;"spaces."
60 NEXT I
70 LPRINT CHR$(27);" ";CHR$(0);
80 LPRINT "This line is Normal space."
90 END
```

```
This   line   is   added   10   spaces.
This   line   is   added   8   spaces.
This   line   is   added   6   spaces.
This   line   is   added   4   spaces.
This   line   is   added   2   spaces.
This  line  is  added  Normal  space.
```

This command can be used to produce micro-justification, which is a method of justifying lines by increasing the space between each character.

■ Printing at half-speed
This printer can print at the fine of 180 characters per second (CPS) with the draft characters or 60 CPS with the LQ characters. But it will also more slowly if you want it to: the half-speed mode prints at 90 CPS with the draft characters or 30

CPS with the LQ characters. The main advantage to half-speed printing is a quieter run.

■ Uni-directional printing

Uni-directional printing means *printing in one direction only*. Your printer normally prints when the print head is moving in both directions. But once in a while you may have an application such as making table where you are more concerned with how the vertical lines align than with how fast it prints. The printer lets you make this choice. The table below shows the commands for controlling how the printer prints.

Table 6-3
Printing direction commands

Function	Control code
Print in one direction	⟨ESC⟩ "U" 1
Print in both directions	⟨ESC⟩ "U" 0
One time print in one direction	⟨ESC⟩ "⟨"

Try this program to see the difference that printing in one direction makes.

```
10 ' Demo of uni-directional printing
20 LPRINT CHR$(27);"A";CHR$(6);
30 FOR I=1 TO 10
40 LPRINT "¦"
50 NEXT I
60 LPRINT :LPRINT
70 LPRINT CHR$(27);"U1";
80 FOR I=1 TO 10
90 LPRINT "¦"
100 NEXT I
110 LPRINT CHR$(12);CHR$(27);"@"
```

Here is what you will get. The top line is printed bi-directionally, and the bottom is printed uni-directionally. You will have to look hard because there isn't much difference, but the bottom vertical lines are perfectly aligned.

Let's analyze the program. Line 20 defines the line spacing to 6/60 of an inch so that the characters that we print will touch top to bottom. Lines 30 ∼ 50 print 10 vertical line characters. Then line 70 sets uni-directional printing and the vertical lines are printed again. Finally line 110 sends a form feed to advance the paper to the top of a new page, and then uses the master reset to restore the printer to the power on condition.

You can also set the printer to print in one direction for one line only by using the ⟨ESC⟩ "⟨" command. This command immediately moves the print head to the left margin and then prints the remainder of the line from left to right.

■ The seven bit dilemma

Some computers (not the IBM-PC fortunately!) don't have the capability to send eight bits on their parallel interface, but can only send seven bits. This would make it impossible for these computers to use this printer's block graphics characters and special symbols if our engineers hadn't thought of a solution. (All of these characters have ASCII codes greater than 127 which means that the eighth bit must be on to use them.) The solution lies in the three control codes given in the following table:

Table 6-4
Eighth bit controls

Function	Control code
Turn the eighth bit ON	⟨ESC⟩ ">" (except IBM-P mode)
Turn the eighth bit OFF	⟨ESC⟩ "=" (except IBM-P mode)
Accept the eighth bit "as is" from the computer	⟨ESC⟩ "#" (except IBM-P mode)

■ Block graphics characters and special symbols

Besides the upper and lower case letters and symbols that we are by now familiar with, your printer has a whole different set of characters that are for special uses. These characters include block graphics for drawing forms and graphs, and special symbols for mathematical, engineering and professional uses. The special characters are included in two character sets. The character set you normally use with the IBM modes is called character set #1. The special characters are printed out when you send ASCII codes 160 ~ 255 to the printer.

Your printer also offers character set #2 which is almost the same as character set #1 except for the addition of ASCII codes 3 ~ 6, 21, and 128 ~ 159. Character set #2 is selected with ⟨ESC⟩ "6"; to go back to character set #1, use ⟨ESC⟩ "7".

You can also specify the power-on default character set by setting DIP switch 1-5 on for character set #1 and off for character set #2 while the IBM modes are selected. The following program will print out all of the graphics characters available:

```
10 LPRINT CHR$(27);"0";
20 LPRINT CHR$(27);"6";
30 FOR J=3 TO 6
40 LPRINT "  ";J;CHR$(J);CHR$(9);
50 NEXT J
60 LPRINT "  21 ";CHR$(21)
70 LPRINT
80 FOR J=128 TO 254 STEP 5
90 FOR I=J TO J+4
100 IF I>254 THEN 140
110 LPRINT I;CHR$(I);CHR$(9);
120 NEXT I
130 LPRINT :LPRINT
140 NEXT J
```

3 ♥	4 ♦	5 ⚊	6 ♠	21 §
128 Ç	129 ü	130 é	131 â	132 ä
133 à	134 å	135 ç	136 ê	137 ë
138 è	139 ï	140 î	141 ì	142 Ä
143 Å	144 É	145 æ	146 Æ	147 ô
148 ö	149 ò	150 û	151 ù	152 ÿ
153 Ö	154 Ü	155 ¢	156 £	157 ¥
158 ₧	159 ƒ	160 á	161 í	162 ó
163 ú	164 ñ	165 Ñ	166 ª	167 º
168 ¿	169 ⌐	170 ¬	171 ½	172 ¼
173 ¡	174 «	175 »	176 ▒	177 ▓
178 █	179 │	180 ┤	181 ╡	182 ╢
183 ╖	184 ╕	185 ╣	186 ║	187 ╗
188 ╝	189 ╜	190 ╛	191 ┐	192 └
193 ┴	194 ┬	195 ├	196 ─	197 ┼
198 ╞	199 ╟	200 ╚	201 ╔	202 ╩
203 ╦	204 ╠	205 ═	206 ╬	207 ╧
208 ╨	209 ╤	210 ╥	211 ╙	212 ╘
213 ╒	214 ╓	215 ╫	216 ╪	217 ┘
218 ┌	219 █	220 ▄	221 ▌	222 ▐
223 ▀	224 α	225 β	226 Γ	227 π
228 Σ	229 σ	230 µ	231 τ	232 Φ
233 Θ	234 Ω	235 δ	236 ∞	237 ∅
238 ∈	239 ∩	240 ≡	241 ±	242 ≥
243 ≤	244 ⌠	245 ⌡	246 ÷	247 ≈
248 °	249 ·	250 -	251 √	252 ⁿ
253 ²	254 ■			

Figure 6-1. Character set #2

Figure 6-1 shows what this program will print. If your chart doesn't look like this because it has regular letters and numbers instead of the special symbols, then your computer is only using seven bits (unless you have set Standard mode by mistake). You can get the correct printout by changing line 110 to this:

```
110 LPRINT I;CHR$(27);">";CHR$(I);CHR$(27);"=";
    CHR$(9);
```

So how are all of these characters used? Here is a program that demonstrate how the graphics characters can be combined to create a figure: the 5 of clubs.

```
10 LPRINT CHR$(27);"6";
20 LPRINT CHR$(218);CHR$(196);CHR$(196);
   CHR$(196);CHR$(196);CHR$(196);CHR$(191)
30 LPRINT CHR$(179);CHR$(53);CHR$(32);CHR$(32);
   CHR$(32);CHR$(32);CHR$(179)
40 LPRINT CHR$(179);CHR$(32);CHR$(5);CHR$(32);
   CHR$(5);CHR$(32);CHR$(179)
50 LPRINT CHR$(179);CHR$(32);CHR$(32);CHR$(5);
   CHR$(32);CHR$(32);CHR$(179)
60 LPRINT CHR$(179);CHR$(32);CHR$(5);CHR$(32);
   CHR$(5);CHR$(32);CHR$(179)
70 LPRINT CHR$(179);CHR$(32);CHR$(32);CHR$(32);
   CHR$(32);CHR$(53);CHR$(179)
80 LPRINT CHR$(192);CHR$(196);CHR$(196);
   CHR$(196);CHR$(196);CHR$(196);CHR$(217)
90 LPRINT CHR$(27);"7"
```

If you have a 7-bit interface, add the following lines to the program given above:

```
15 LPRINT CHR$(27);">";
95 LPRINT CHR$(27);"="
```

In this program, line 10 selects character set #2 so that all the graphics characters can be used (including the "club" symbol). Then lines 20 ∼ 80 print the 5 of clubs figure. Line 90 cancels character set #2 (which is the same as selecting character set #1).

Here is what this program prints:

```
5
  ♣  ♣
    ♣
  ♣  ♣
        5
```
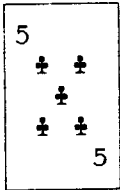
■ International character sets

Your printer changes languages by changing 12 characters that are different in the different languages. These sets of characters are called *international character sets* — and there are 13 sets in all. The control codes to select the international character sets are given in Table 6-5.

Table 6-5
International character set commands

Country	Control code
U.S.A.	⟨ESC⟩ "R" CHR$(0)
France	⟨ESC⟩ "R" CHR$(1)
Germany	⟨ESC⟩ "R" CHR$(2)
England	⟨ESC⟩ "R" CHR$(3)
Denmark type I	⟨ESC⟩ "R" CHR$(4)
Sweden	⟨ESC⟩ "R" CHR$(5)
Italy	⟨ESC⟩ "R" CHR$(6)
Spain type I	⟨ESC⟩ "R" CHR$(7)
Japan	⟨ESC⟩ "R" CHR$(8)
Norway	⟨ESC⟩ "R" CHR$(9)
Denmark type II	⟨ESC⟩ "R" CHR$(10)
Spain type II	⟨ESC⟩ "R" CHR$(11)
Latin America	⟨ESC⟩ "R" CHR$(12)

NOTE: When you set the printer to the IBM-P mode, use ⟨FS⟩ instead of ⟨ESC⟩.

The characters that change are shown beneath their ASCII code in Table 6-6.

Table 6-6
International character sets

Country	35	36	64	91	92	93	94	96	123	124	125	126
U.S.A.	#	$	@	[	\	]	^	`	{	|	}	~
France	#	$	à	°	ç	§	^	`	é	ù	è	¨
Germany	#	$	§	Ä	Ö	Ü	^	`	ä	ö	ü	ß
England	£	$	@	[	\	]	^	`	{	|	}	~
Denmark type I	#	$	@	Æ	Ø	Å	^	`	æ	ø	å	~
Sweden	#	¤	É	Ä	Ö	Å	Ü	é	ä	ö	å	ü
Italy	#	$	@	°	\	é	^	ù	à	ò	è	ì
Spain	₧	$	@	¡	Ñ	¿	^	'	¨	ñ	}	~
Japan	#	$	@	[	¥	]	^	`	{	|	}	~
Norway	#	¤	É	Æ	Ø	Å	Ü	é	æ	ø	å	ü
Denmark type II	#	$	É	Æ	Ø	Å	Ü	é	æ	ø	å	ü
Spain type II	#	$	á	¡	Ñ	¿	é	`	í	ñ	ó	ú
Latin America	#	$	á	¡	Ñ	¿	é	ü	í	ñ	ó	ú

■ Printing characters in the control code area

When you refer the Appendix C, you'll find many characters are printed in the control code area. (Remember that the low-order control codes are the ASCII codes 0 through 31 plus 127, and the high-order control codes are 128 through 159 plus 255.)

These codes don't normally print symbols on paper, rather they cause the printer to change modes. To make them print as normal symbols requires an extra command. For example, the command to "normalize" the high-order control codes is ⟨ESC⟩ "6", which is the same as selecting the Character set #2.

The low-order control codes can also be defined with the IBM-P mode. Try this program with the IBM-P mode.

```
10 ' Demo of characters in IBM-P mode
20 LPRINT CHR$(27);"\";CHR$(32);CHR$(0);
30 FOR I=0 TO 31
40 LPRINT CHR$(I);
50 NEXT I
60 LPRINT
70 END
```

In this program line 20 determines that the following 32 characters should be printed as printable characters. So, the characters in the control code area are printed as shown below.

♥♦±♠   ¶§   →←   Ø

If you want to print more characters using the ⟨ESC⟩ "\" command you must tell the printer how many characters should be printed. The total number of characters is equal to *n1* + (*n2* × 256).

For example, if you want to print 300 characters, *n1* = 44 and *n2* = 1 (300 = 44 + 1 × 256), the control code will be the following:

⟨ESC⟩ "\" CHR$(44) CHR$(1)

There is one more command to print a character in the control code area: ⟨ESC⟩ "ˆ" *n*. This command prints one character determined by the value of *n*. You'll get the same results as shown above with the following program.

```
10 ' Demo of one character print
20 FOR I=0 TO 31
30 LPRINT CHR$(27);"^";
40 LPRINT CHR$(I);
50 NEXT I
60 LPRINT
70 END
```

A note for IBM-PC users:

When you run these programs, you cannot get the right-pointed arrow (CHR$(26)) with the IBM-PC computer. This is because the IBM-PC does not send this code to the printer.

To avoid this problem, change line 40, and add line 100 and 120 to either of the following lists.

```
40 GOSUB 100
100 O=INP(&H379)  : IF O<128 THEN 100
```

```
110 OUT &H378,I   : OUT &H37A,5   : OUT &H37A,4
120 RETURN

40 GOSUB 100
100 O=INP(&H3BD)  : IF O<128 THEN 100
110 OUT &H3BC,I   : OUT &H3BE,5   : OUT &H3BE,4
120 RETURN
```

■ Printing BIG characters

You can even enlarge your character sets for attention-grabbing headings or special effects. There are two commands you can use. Everything following any of them will be enlarged as shown below, until the cancel code is entered.

Table 6-7
Big character commands

Function	Control code
Double-high enlarged print	⟨ESC⟩ "h" CHR$(1)
Quad-high enlarged print	⟨ESC⟩ "h" CHR$(2)
Cancel enlarged print	⟨ESC⟩ "h" CHR$(0)

Try this program to see the big characters.

```
10 ' Demo of big characters
20 LPRINT "THIS IS ";
30 LPRINT CHR$(27);"h";CHR$(1);
40 LPRINT "DOUBLE";
50 LPRINT CHR$(27);"h";CHR$(0);
60 LPRINT " SIZED PRINTING."
70 LPRINT
80 LPRINT "THIS IS ";
90 LPRINT CHR$(27);"h";CHR$(2);
100 LPRINT "QUAD";
110 LPRINT CHR$(27);"h";CHR$(0);
120 LPRINT " SIZED PRINTING."
130 END
```

When you run this program, you will get like this:

THIS IS DOUBLE SIZED PRINTING.

THIS IS QUAD SIZED PRINTING.

As you can see, when the big character command is used, the baseline for each character does not align. If you want to align the baseline, try this program:

```
10 ' Demo of aligning big characters
20 LPRINT "THIS IS ";
30 LPRINT CHR$(27);"j";CHR$(18);
40 LPRINT CHR$(27);"h";CHR$(1);
50 LPRINT "DOUBLE";
60 LPRINT CHR$(27);"h";CHR$(0);
70 LPRINT CHR$(27);"J";CHR$(18);
80 LPRINT " SIZED PRINTING."
90 LPRINT :LPRINT :LPRINT
100 LPRINT "THIS IS ";
110 LPRINT CHR$(27);"j";CHR$(54);
120 LPRINT CHR$(27);"h";CHR$(2);
130 LPRINT "QUAD";
140 LPRINT CHR$(27);"h";CHR$(0);
150 LPRINT CHR$(27);"J";CHR$(54);
160 LPRINT " SIZED PRINTING."
170 END
```

When you run this program, you will get this:

THIS IS DOUBLE SIZED PRINTING.

THIS IS QUAD SIZED PRINTING.

■ The optional sheet feeder
The automatic sheet feeder is a handy option that feeds single

cut sheets automatically. Work done on cut sheets looks better that done on computer paper, and you don't have to tear the "perforations" off each sheet as you must with fan-fold paper.

The automatic sheet feeder feeds a new sheet automatically every time the printer receives or generates a form feed. Any time you wish, you can turn the auto-feed unit on and off by using control codes.

Table 6-8
Automatic sheet feeder commands

Function	Control code
Select automatic feed mode	⟨ESC⟩ CHR$(25) CHR$(4) *or* "((4))"
Cancel automatic feed mode	⟨ESC⟩ CHR$(25) CHR$(0) *or* "((0))"
Insert paper from first bin	⟨ESC⟩ CHR$(25) CHR$(1) *or* "((1))"
Insert paper from second bin	⟨ESC⟩ CHR$(25) CHR$(2) *or* "((2))"
Eject paper	⟨ESC⟩ CHR$(25) "R" *or* "((R))"
Set print start position	⟨ESC⟩ CHR$(25) "T" *n* or "((T))" *n*

When the automatic sheet feeder is installed, you must set DIP switch 2-4 on to detect the paper-out condition.

In addition, the following functions are ignored when the automatic sheet feeder is installed:

Setting of the page length
Top and bottom margins
Vertical tab settings
Reverse feed functions

When you want to change the print start position, try to use the ⟨ESC⟩ CHR$(25) "T" *n* command.

The value of *n* determines the print start position from the top edge of the each page. For example,⟨ESC⟩ CHR$(25) "T" "1" command determines to print from the top edge of the paper.

■ Reading a hex dump

The BASIC in some computers changes ASCII codes before they send them to the printer. If you run into problem because of this, try this hex dump to check the ASCII codes.

First turn off the printer. Hold down both the Paper Feed key and Quality key and turn on the printer, and run the following program.

```
10 FOR I=0 TO 255
20 LPRINT CHR$(I);
30 NEXT I
40 LPRINT
50 END
```

If your system passes the codes directly to the printer without changing them, you will get a printout like this. (You can print out the last remaining line in the print buffer by putting the printer off line with the On Line key.)

```
00 01 02 03 04 05 06 07    08 09 0A 0B 0C 0D 0E 0F    . . . . . . . .  . . . . . . . .
10 11 12 13 14 15 16 17    18 19 1A 1B 1C 1D 1E 1F    . . . . . . . .  . . . . . . . .
20 21 22 23 24 25 26 27    28 29 2A 2B 2C 2D 2E 2F     !"#$%&'  ()*+,-./
30 31 32 33 34 35 36 37    38 39 3A 3B 3C 3D 3E 3F    01234567 89:;<=>?
40 41 42 43 44 45 46 47    48 49 4A 4B 4C 4D 4E 4F    @ABCDEFG HIJKLMNO
50 51 52 53 54 55 56 57    58 59 5A 5B 5C 5D 5E 5F    PQRSTUVW XYZ[\]^_
60 61 62 63 64 65 66 67    68 69 6A 6B 6C 6D 6E 6F    `abcdefg hijklmno
70 71 72 73 74 75 76 77    78 79 7A 7B 7C 7D 7E 7F    pqrstuvw xyz{|}~.
80 81 82 83 84 85 86 87    88 89 8A 8B 8C 8D 8E 8F    . . . . . . . .  . . . . . . . .
90 91 92 93 94 95 96 97    98 99 9A 9B 9C 9D 9E 9F    . . . . . . . .  . . . . . . . .
A0 A1 A2 A3 A4 A5 A6 A7    A8 A9 AA AB AC AD AE AF     !"#$%&'  ()*+,-./
B0 B1 B2 B3 B4 B5 B6 B7    B8 B9 BA BB BC BD BE BF    01234567 89:;<=>?
C0 C1 C2 C3 C4 C5 C6 C7    C8 C9 CA CB CC CD CE CF    @ABCDEFG HIJKLMNO
D0 D1 D2 D3 D4 D5 D6 D7    D8 D9 DA DB DC DD DE DF    PQRSTUVW XYZ[\]^_
E0 E1 E2 E3 E4 E5 E6 E7    E8 E9 EA EB EC ED EE EF    `abcdefg hijklmno
F0 F1 F2 F3 F4 F5 F6 F7    F8 F9 FA FB FC FD FE FF    pqrstuvw xyz{|}~.
0D 0A                                                 . .
```

Most BASICs, however, are not quite that straight forward. For example, the IBM-PC prints the following:

```
00 01 02 03 04 05 06 07    08 09 0A 0B 0C 0D 0A 0E    . . . . . . . .  . . . . . . . .
0F 10 11 12 13 14 15 16    17 18 19 1B 1C 1D 1E 1F    . . . . . . . .  . . . . . . . .
20 21 22 23 24 25 26 27    28 29 2A 2B 2C 2D 2E 2F     !"#$%&'  ()*+,-./
30 31 32 33 34 35 36 37    38 39 3A 3B 3C 3D 3E 3F    01234567 89:;<=>?
40 41 42 43 44 45 46 47    48 49 4A 4B 4C 4D 4E 4F    @ABCDEFG HIJKLMNO
50 51 52 53 54 55 56 57    58 59 5A 5B 5C 5D 5E 5F    PQRSTUVW XYZ[\]^_
60 61 62 63 64 65 66 67    68 69 6A 6B 6C 6D 6E 6F    `abcdefg hijklmno
70 71 72 73 74 75 76 77    78 79 7A 7B 7C 7D 7E 7F    pqrstuvw xyz{|}~.
80 81 82 83 84 85 86 87    88 89 8A 8B 8C 8D 8E 8F    . . . . . . . .  . . . . . . . .
90 91 92 93 94 95 96 97    98 99 9A 9B 9C 9D 9E 9F    . . . . . . . .  . . . . . . . .
A0 A1 A2 A3 A4 A5 A6 A7    A8 A9 AA AB AC AD AE AF     !"#$%&'  ()*+,-./
B0 B1 B2 B3 B4 B5 B6 B7    B8 B9 BA BB BC BD BE BF    01234567 89:;<=>?
C0 C1 C2 C3 C4 C5 C6 C7    C8 C9 CA CB CC CD CE CF    @ABCDEFG HIJKLMNO
D0 D1 D2 D3 D4 D5 D6 D7    D8 D9 DA DB DC DD DE DF    PQRSTUVW XYZ[\]^_
E0 E1 E2 E3 E4 E5 E6 E7    E8 E9 EA EB EC ED EE EF    `abcdefg hijklmno
F0 F1 F2 F3 F4 F5 F6 F7    F8 F9 FA FB FC FD FE FF    pqrstuvw xyz{|}~.
0D 0A                                                 . .
```

Notice that your printer receives decimal code 13 (hex 0D) together with hex 0A, which is really decimal 10. In addition, your printer does not receive decimal code 26 (hex 1A).

Your printer prints 16 hex numbers per line, printing the characters on the right hand side. If it receives less than 16, it sits in a holding pattern, awaiting more data. Taking the printer off line dumps the characters to the paper.

To debug a program quickly, just use the hex dump capability. Appendix B will help you translate the hex codes to ASCII equivalents.

MEMO

CHAPTER 7
DOWNLOAD CHARACTERS AND DOT GRAPHICS

Subjects covered in Chapter 7 include—
- **Designing and printing your own characters**
- **User-designed graphics**

Using the procedures outlined so far, all of the printer's standard functions can be utilized. There are two additional user-defined functions available: designing and printing your own characters, and designing and printing your own dot graphics.

These two functions can be used to design and print any symbol, or any graphic shape. This allows you to make your own special symbols, or print a company logo. Some understanding of dot matrix technology will be useful.

DOT MATRIX PRINTING

Before you start to learn how to create your own characters, you'll need some understanding of how dot matrix printers work. The process is called dot-matrix printing because each character is composed of small dots arranged on a matrix, or grid. To see how that character grid gets printed, you need to take a look at the print head itself.

The print head in this printer comprises 24 wires apparently arranged in a vertical column. If you were to remove the print head and look at the wires, you would see two staggered columns of twelve wires. However, the timing is such that they print as one continuous column of 24 wires.

Figure 7-1 shows an enlarged schematic view of the front of

the print head, showing the ends of the wires and their relationship to the draft characters. As you can see, the capital letters use the top eighteen wires of the print head, and the descenders (such as the lower case "p" shown) use the bottom eighteen pins. As the print head moves across the page (in either direction — that's what is meant by bi-directional printing) it prints one column of dots at a time. Each time a dot is supposed to print an electromagnet inside the print head causes the appropriate wire to strike the ribbon (making this printer an impact printer).
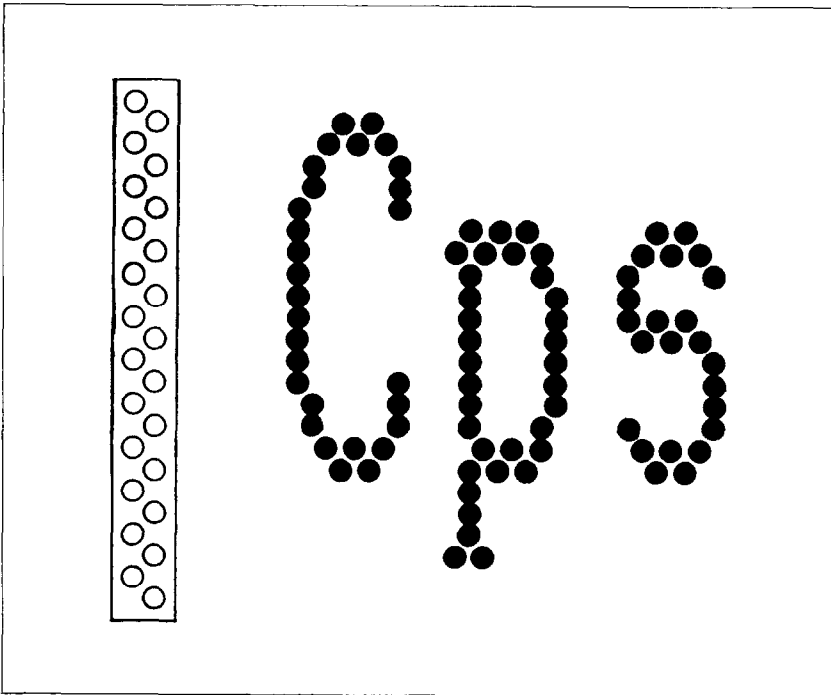

Figure 7-1. As the print head moves across the page, each of the wires prints one row of dots.

DESIGNING YOUR OWN CHARACTERS

Standard characters are permanently stored in the printer's ROM (Read Only Memory), but characters you design are

downloaded and stored in RAM (Random Access Memory) for use.

Designing and printing your own characters has two requirements: firstly, designing the shape of the character, calculating the data necessary to make the shape, and sending that data to the printer, and secondary, sending the command to print the downloaded characters instead of the standard characters. There are a number of design constraints for download draft quality characters:

- The matrix or grid on which you design the characters is five boxes wide by twenty-four boxes high.
- Horizontally, dots may be centered in a box, or may straddle a line, making the actual character grid 9 dots wide by 24 dots high. Vertically, dots can only be centered in a box. See Figure 7-2.
- Dots cannot overlap — that is, you may not have a dot inside a box next to one that is on a line.
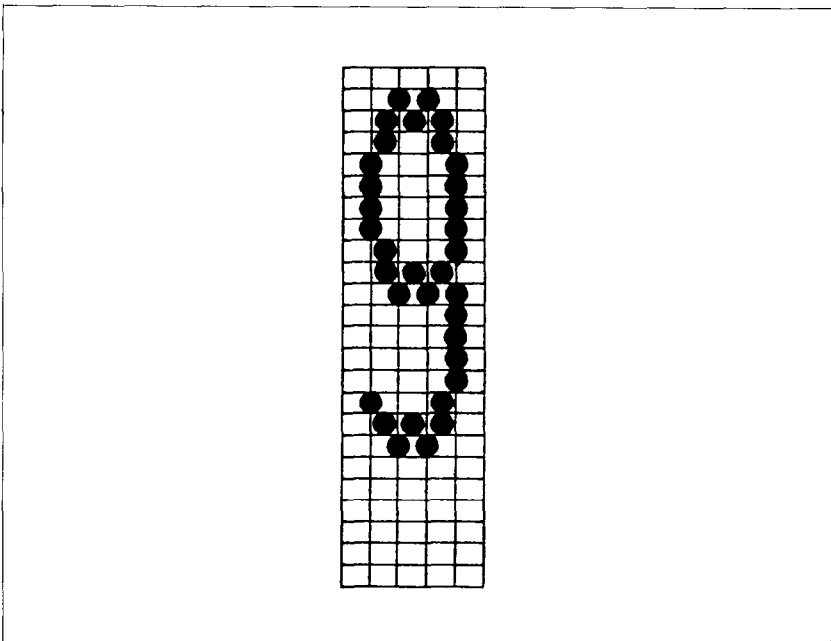

Figure 7-2. Dots can be inside boxes or straddle the vertical lines of the grid.

- You may define any position in the ASCII table except the control code area, but you cannot define more than 35 characters.

Photocopy the grid in Figure 7-3 to help design your new characters. We will use a tiny representation of a picture of an automobile for our example.

Next we calculate the *vertical* numerical values of the columns of dots, and enter them underneath the grid. Looking at Figure 7-4, we see that each vertical column (which has a maximum of 24 dots) is first divided into three groups of eight dots. Each group of eight dots is represented by one byte, which consists of eight bits. For example, looking at Figure 7-4, we see that in the left-most column there are five dots, and they are sitting in the "1" box of the 1st byte, "128", "64", "32", and "16" boxes of the 2nd byte. Thus its vertical value is 1 for the 1st byte, 240 for the 2nd byte, and 0 for the 3rd byte.
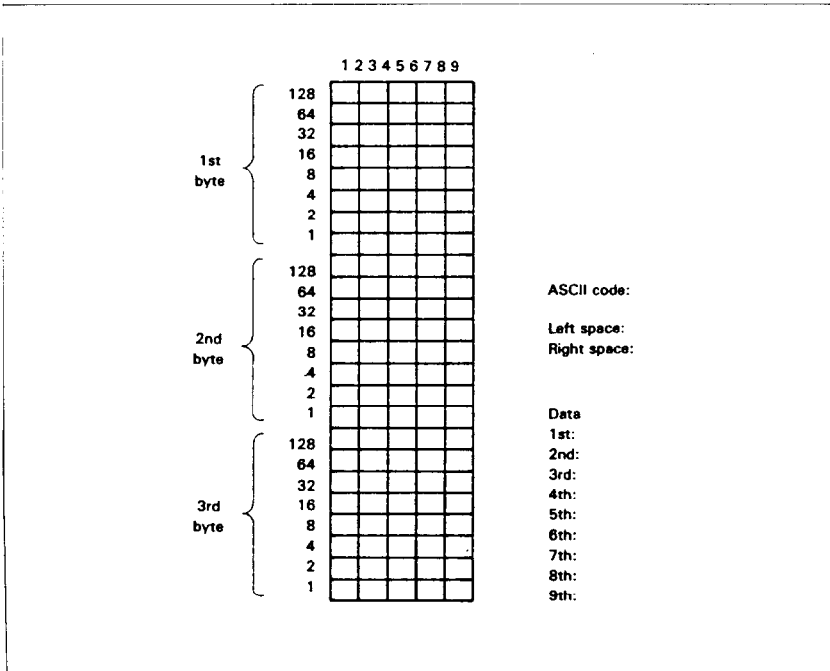

Figure 7-3. Use this grid (or one similar to it) to define your own draft characters.
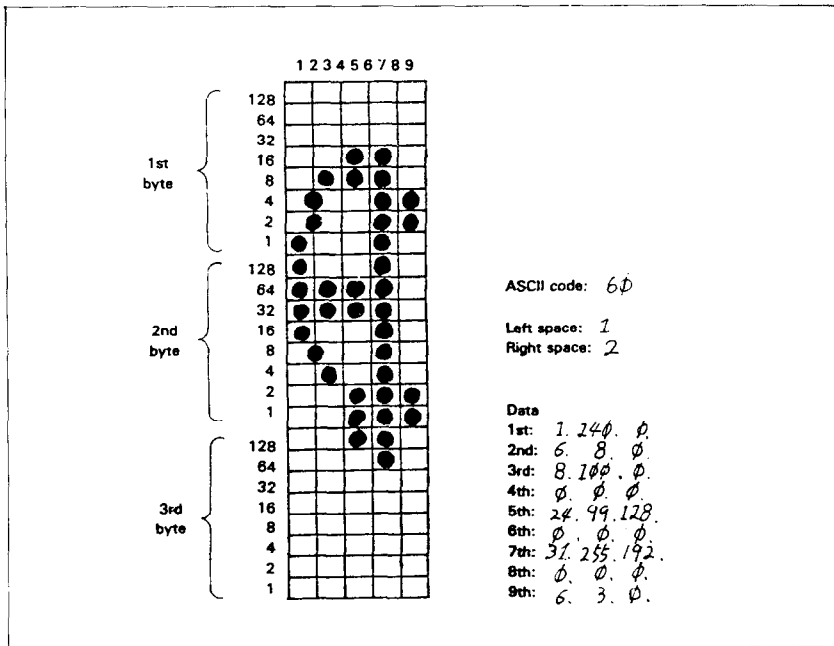
Figure 7-4. Add the values of the dots in each column and write the sum of each column at the bottom.

■ Assigning a value of character space

Besides being able to specify the actual width of the character, this printer allows you to specify the position in the standard grid where the character will print. You must specify the dot column in which the printed character starts and the dot column in which the character ends. Why, you may ask, would you want to define a character this way instead of merely defining the overall width of the character? Because this printer's draft proportional character definitions can also be used to print normal width characters, which are twelve dot columns wide. And by centering even the narrow characters in the complete grid they will look good even you aren't printing them proportionally.

The three bytes are used to specify the width of the character and the space to be allowed on either side of it. The left space (in dot columns) is specified by *m0* and the right space is specified by *m2*. The second byte (*m1*) specifies the number of columns of dots that will be printed by the character. By varying the width

of the character itself and the spaces around it, you can actually create proportional width characters that print at draft speed.

When defining draft characters, the number of printed columns ($m1$) cannot exceed 9, and the sum of $m0 + m1 + m2$ cannot exceed 12.

■ Assigning a value to your character

We've done a pretty through job of designing and describing a user-defined character. But this printer has room for 35 download characters — how does it knows which standard character we want to print: every character is assigned a unique number.

The standard characters are assigned the ASCII code — numbers from 0 to 255. For the download character sets you can define any positions between 32 to 127 and between 160 to 255. This means that once a character is defined and assigned a value (and the download character set is selected), you can use that character on the printer the same way you would any standard
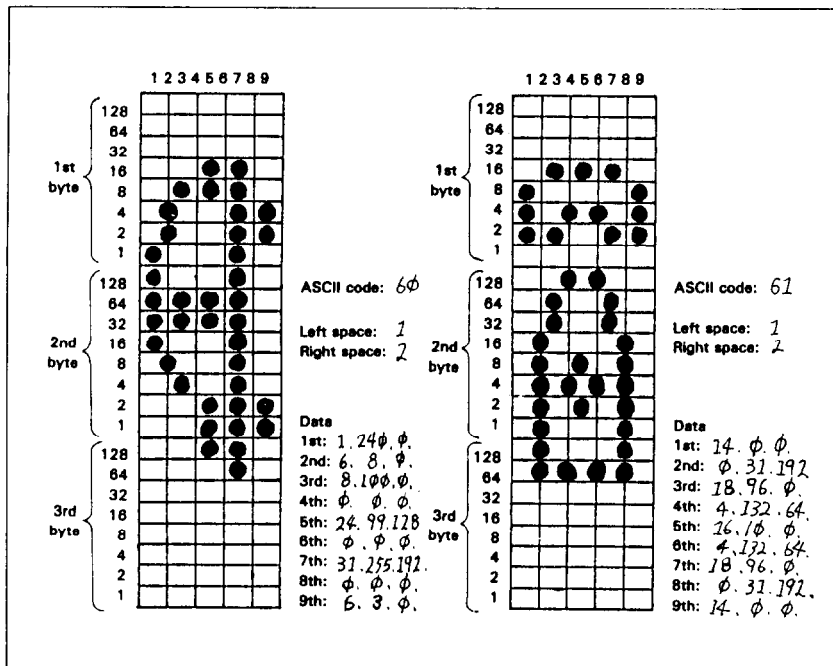

Figure 7-5. Character designs for the two graph symbols.

character. You can send the character with the same ASCII value. You can also access the character from a BASIC program with the CHR$ function.

In our example, we'll assign the car a value of 60, which is the code for the character "⟨" in the ASCII characters. A rather arbitrary selection, but this printer doesn't care!

Our chart would hardly be complete with just a picture of a car, so in Figure 7-5 we've made completed grids for another symbol: a telephone (quite a strange mix of characters!). The information on the grids is now completed.

■ Download character definition command

You've read through a long explanation of download characters and we haven't even told you the command syntax yet! Now the wait is over. This is the most complex command in your printer repertoire and now you've got the necessary knowledge to implement it. Here it is:

⟨ESC⟩ "&" CHR$(0) *n1 n2 m0 m1 m2 d1 d2 d3 ... dx*
<div style="text-align: right">(except for the IBM-P mode)</div>

⟨ESC⟩ "=" CHR$(0) *n1 n2 m0 m1 m2 d1 d2 d3 ... dx*
<div style="text-align: right">(only for the IBM-P mode)</div>

Like the other printer's commands, it starts with an ⟨ESC⟩ (CHR$(27)). The next character is an ampersand (&) (CHR$(38)) or an equal (=) (CHR$(61)) followed by a CHR$(0).

n1 and *n2* are used to specify the ASCII values of the characters you are defining. The reason that there are two bytes reserved for this is that your printer allows you to define many characters with just a single command. *n1* is used to specify the beginning of a range of characters to be defined; *n2* specifies the end of the range. For instance, if you wanted to change the appearance of the numerals from 0 to 9 (which have ASCII codes 48 through 57), the command would begin with ⟨ESC⟩ "&" CHR$(0) CHR$(48) CHR$(57) .. (or ⟨ESC⟩ "=" CHR$(0) CHR$(48) CHR$(57) ..). Of course, you can also define individual characters by making *n1* and *n2* equal.

The three bytes following the specification of the range of

characters (*m0*, *m1* and *m2*) are used to specify the width of the character and the space to be allowed on either side of it. The left space (in dot column) is specified by *m0* and the right space is specified by *m2*. The second byte (*m1*) specifies the number of columns of dots that will be printed by the character. By varying the width of the character itself and the spaces around it, you can actually create proportional width characters.

You'll probably recognize *d1 d2 ... dx* from the top of our layout grid. That's right, each column is described by three bytes. Now we've got everything we need to download one character to the printer. The complete command for our car character with the Standard mode is shown below:

```
CHR$(27);CHR$(38);CHR$(0);CHR$(60);CHR$(60);
CHR$(1);CHR$(9);CHR$(2);CHR$(1);CHR$(240);CHR$(0);
CHR$(6);CHR$(8);CHR$(0);CHR$(8);CHR$(100);CHR$(0);
CHR$(0);CHR$(0);CHR$(0);CHR$(24);CHR$(99);CHR$(128);
CHR$(0);CHR$(0);CHR$(0);CHR$(31);CHR$(255);CHR$(192);
CHR$(0);CHR$(0);CHR$(0);CHR$(6);CHR$(3);CHR$(0)
```

Now let's send the information to the printer. The following program will send the character definitions for all two characters to the printer. Before you enter the program, turn off the printer and set the DIP switch 2-3 off. Otherwise, your printer will ignore your download definitions.

After you set the DIP switch correctly, enter the program and run it.

```
10 LPRINT CHR$(27);"x0";
20 LPRINT CHR$(27);"&";CHR$(0);CHR$(60);CHR$(61);
30 FOR N=60 TO 61
40 FOR M=1 TO 30
50 READ MM
60 LPRINT CHR$(MM);
70 NEXT M
80 NEXT N
90 LPRINT
100 END
110 DATA    1,  9,  2,  1,240,  0,  6,  8,  0,  8,
     100,  0
```

```
120 DATA   0,  0,  0, 24, 99,128,  0,  0,  0, 31,
    255,192
130 DATA   0,  0,  0,  6,  3,  0
140 DATA   1,  9,  2, 14,  0,  0,  0, 31,192, 18,
     96,  0
150 DATA   4,132, 64, 16, 10,  0,  4,132, 64, 18,
     96,  0
160 DATA   0, 31,192, 14,  0,  0
```

When you run this program, it looks like nothing happens. That's OK. We'll see why in just a moment. Save this program. We'll need it again shortly.

PRINTING DOWNLOAD CHARACTERS

You've now defined and sent two characters to your printer. But how do you know that? If you try printing those characters now you don't get a car and telephone. Instead you get .. ⟨=. That's because the download characters are stored in a different part of the printer's memory. To tell it to look in download character RAM instead of standard character ROM it requires another command:

⟨ESC⟩ "%" n

Where n is either 1 (to select), or 0 (to cancel). If you want your new characters to be included in a standard character set, you may copy the standard set stored in ROM into the download RAM, which will save you switching back and forth between the download area and the standard ROM. If you wish to use this option, make sure that you copy the standard characters into RAM **before** you download your own, because the last-copied characters replace previously copied ones. The command to copy the standard characters into the download RAM area is:

⟨ESC⟩ ":" CHR$(0) CHR$(0) CHR$(0)

(except for the IBM-P mode)

⟨FS⟩ ":" CHR$(0) CHR$(0) CHR$(0)

(only for the IBM-P mode)

To demonstrate how to use the download characters, let's use the "car" and "telephone" user-defined characters to print a small graph. This program will do just that:

```
10 LPRINT CHR$(27);"x0";
20 LPRINT CHR$(27);":";CHR$(0);CHR$(0);CHR$(0)
30 LPRINT CHR$(27);"&";CHR$(0);CHR$(60);CHR$(61);
40 FOR N=60 TO 61
50 FOR M=1 TO 30
60 READ MM
70 LPRINT CHR$(MM);
80 NEXT M
90 NEXT N
100 LPRINT
110 DATA    1,   9,   2,   1,240,   0,   6,   8,   0,   8,
   100,   0
120 DATA   0,   0,   0,  24,  99,128,   0,   0,   0,  31,
   255,192
130 DATA   0,   0,   0,   6,   3,   0
140 DATA    1,   9,   2,  14,   0,   0,   0,  31,192,  18,
    96,   0
150 DATA    4,132,  64,  16,  10,   0,   4,132,  64,  18,
    96,   0
160 DATA   0,  31,192,  14,   0,   0
170 LPRINT CHR$(27);"D";CHR$(11);CHR$(0)
180 LPRINT CHR$(27);"h";CHR$(1);
190 LPRINT " DIFFUSION RANGES OF"
200 LPRINT "  CARS & TELEPHONES"
210 LPRINT CHR$(27);"h";CHR$(0);
220 LPRINT CHR$(27);"%1";
230 LPRINT "USA";CHR$(9);
240 FOR I=0 TO 681 STEP 50
250 LPRINT CHR$(60);
260 NEXT I
270 LPRINT
280 LPRINT CHR$(9);
290 FOR I=0 TO 781 STEP 50
300 LPRINT CHR$(61);
```

```
310 NEXT I
320 LPRINT
330 LPRINT "GERMANY";CHR$(9);
340 FOR I=0 TO 412 STEP 50
350 LPRINT CHR$(60);
360 NEXT I
370 LPRINT
380 LPRINT CHR$(9);
390 FOR I=0 TO 488 STEP 50
400 LPRINT CHR$(61);
410 NEXT I
420 LPRINT
430 LPRINT "JAPAN";CHR$(9);
440 FOR I=0 TO 347 STEP 50
450 LPRINT CHR$(60);
460 NEXT I
470 LPRINT
480 LPRINT CHR$(9);
490 FOR I=0 TO 493 STEP 50
500 LPRINT CHR$(61);
510 NEXT I
520 LPRINT
530 LPRINT CHR$(9);";+-";
540 SCALE$=;"+-+-"
550 FOR I=2 TO 8 STEP 2
560 LPRINT SCALE$;
570 NEXT I
580 LPRINT "+-+"
590 LPRINT CHR$(9);"   ";
600 FOR I=2 TO 8 STEP 2
610 LPRINT " ";I;
620 NEXT I
630 LPRINT CHR$(27);"%0"
640 LPRINT CHR$(27);"S0";
650 LPRINT CHR$(9);"100 UNITS/1000 PERSONS"
660 LPRINT CHR$(27);"T"
```

```
DIFFUSION  RANGES  OF
  CARS  &  TELEPHONES
USA            ꞯꞯꞯꞯꞯꞯꞯꞯꞯꞯꞯꞯꞯ
               ☏☏☏☏☏☏☏☏☏☏☏☏☏☏
GERMANY        ꞯꞯꞯꞯꞯꞯꞯꞯꞯ
               ☏☏☏☏☏☏☏☏☏☏
JAPAN          ꞯꞯꞯꞯꞯꞯꞯ
               ☏☏☏☏☏☏☏☏☏
               +--+--+--+--+--+--+--+--+--+
                  2    4    6    8
               100  UNITS/1000  PERSONS
```

THE PRINT MODE AND THE DOWNLOAD CHARACTERS

In the example program above, before we defined the new characters, we selected the Draft print mode. By doing this, we caused user-defined characters to print as draft characters because user-defined characters assume the mode that is in effect at the time they are defined.

The modes that affect user-defined characters are draft, and letter quality printing. (The ⟨ESC⟩ "x" n command selects between draft ($n = 0$) and letter quality ($n = 1$).)

The commands for defining characters, selecting the download character set, and copying ROM are exactly the same for each of the user-defined print styles. The print mode in effect when the characters are defined determines the mode of the user-defined characters.

The print mode determines the size of the grid upon which the characters are designed and the speed at which they print. Table 7-1 shows the different character grid sizes for each of the print modes.

Table 7-1
Download character design grids

	$m1$ (Maximum)	$m0 + m1 + m2$ (Maximum)
Draft characters	9	12
LQ characters with pica pitch	29	36
LQ characters with elite pitch	23	30
LQ characters with proportional pitch	37	42

■ **Draft download characters**

As you've seen, draft characters are designed on a grid which is 9 dots wide by 24 dots high (plus up to three columns of dots for space between characters). Draft download characters print at the same high speed as normal draft characters, even though they may print with more dots and may even be designed for proportional spacing.

■ **Letter quality download characters**

If you select letter quality printing with the ⟨ESC⟩ "x" 1 command or by the control panel, you can design your own characters on a grid which is 29, 23, or 37 dots wide by 24 dots high depending on the print pitch. Each character can be as wide as 36, 30, or 42 dots, including space on either side of the character. The dot columns are spaced closer together horizontally than draft style dot columns.

Unlike draft characters, there are no restrictions on which dots can print. This, coupled with the closer dot spacing, allows you to design characters with higher resolution. The drawback is speed. Normal letter quality characters print more slowly than draft characters; the same is true of LQ download characters.

STORING THE DOWNLOAD CHARACTERS

You can define up to 35 characters with this printer. But, once the printer is turned off, you must redefine those characters again.

By using the optional RAM cartridge, you can extend the download characters up to 192, and the download characters are remained even if the printer is turned off because of the battery in the cartridge.

This is a very convenient function when you use and save your own characters for later use.

In addition, when the write-protect switch of the cartridge is set on after downloaded characters, you can print those characters by the panel operation, without using special control codes.

USER-DESIGNED DOT GRAPHICS

Designing and printing graphics is similar to the procedures

for characters, except that the printed width can range from one to 3264 dots on a 15-inch type printer. Selection of the appropriate line spacing will allow rows of dots to link seamlessly, and in this way you may design and print a whole page of data. In the dot graphics mode, you directly address all dot positions on the page if desired. The graphics mode can print as many as 240 dots per inch compared with the standard resolution of 60 dots per inch (DPI).

The command syntax of the dot graphics mode is:

⟨ESC⟩ "*" *n0 n1 n2 m1 m2* .....

Just like many of the other codes you have learned, the command starts with an escape sequence (⟨ESC⟩ "*" in this case). This is followed by *n0*, which specifies the print density as shown in Table 7-2. But unlike the other codes there can be any number of graphics data bytes following the command. That's where *n1* and *n2* come in; they are used to tell the printer how many columns of graphics data to expect.

Table 7-2
Graphics modes

n0	Mode	Print density
0	8-dot normal density	60 dots/inch
1	8-dot double density	120 dots/inch
2	8-dot double density with double-speed	120 dots/inch
3	8-dot quadruple density	240 dots/inch
4	8-dot semi-double density	80 dots/inch
6	8-dot CRT graphics	90 dots/inch
32	24-dot normal density	60 dots/inch
33	24-dot double density	120 dots/inch
38	24-dot CRT graphics	90 dots/inch
39	24-dot triple density	180 dots/inch
40	24-dot hexa density	360 dots/inch

To figure the value of *n1* and *n2*, you need to work out how wide your image will be. Because you are limited to the largest number that can be sent in one byte (255), the formula to work out *n1* and *n2* is:

if the number of columns is X,

then $n1 = X$ MOD 256, and $n2 = INT(X/256)$

Table 7-3 may make the calculation easier.

Table 7-3
Calculating *nl* and *n2*.

If the number of columns, x, ranges from:	Then *nl* is:	and *n2* is:
1 to 255	x	0
256 to 511	x− 256	1
512 to 767	x− 512	2
768 to 1023	x− 768	3
1024 to 1279	x−1024	4
1280 to 1535	x−1280	5
1536 to 1791	x−1536	6
1792 to 2047	x−1792	7
2048 to 2303	x−2048	8
2304 to 2559	x−2304	9
2560 to 2815	x−2560	10
2816 to 3071	x−2816	11
3072 to 3264	x−3072	12

When you are using the 24-dot graphics modes you must send three bytes of data for each dot column, as you've done with the download characters. Therefore, you refer to dot columns instead of bytes of graphics data when calculating *nl* and *n2*.

■ Specifying the graphics data

Now that we've told the printer how much data to expect, we better figure out how to send that information! Just as you do with download characters, with dot graphics you have control over firing of every single pin of the print head. When this printer produces 8-dot graphics modes, it prints with every third pin with the Standard mode. It acts like a printer with nine wires that can only produce 8-dot graphics except the vertical length. If you want to align the vertical length, use the IBM modes. With this mode, we've designed to align fully compatible with the IBM Graphics printer or Proprinter. In Figure 7-6, you can see that we've labelled each pin on the print head with a number as we did with download characters. And specifying pins to fire is done in the same way: to fire the second pin from the top, for instance, send a CHR$(64). Firing several pins at once is done in a similar fashion. For example, to print the first,

third, and fourth dots, add their values (128 + 32 + 16) to send this total: CHR$(176). This is one byte of graphics data; it would replace *m1* in our format statement.
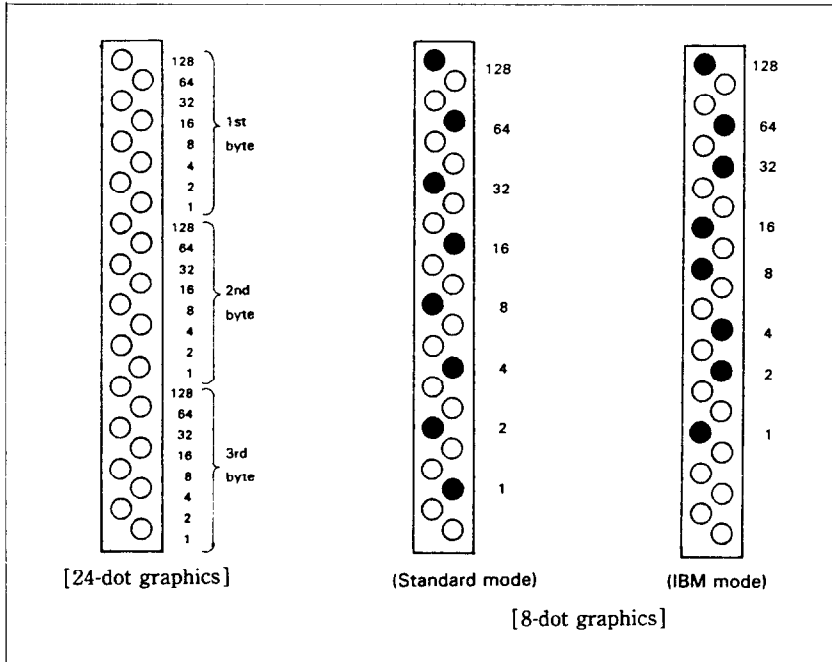

[24-dot graphics]   (Standard mode)   (IBM mode)

[8-dot graphics]

Figure 7-6. Starting with the most significant bit at the top, each third pin of the print head is assigned a value which is a power of two in case of 8-dot graphics modes.

COMPATIBILITY WITH EXISTING SOFTWARE

With its ability to print eleven different graphics densities, this printer's graphics abilities are advanced indeed. There are many programs, in fact, that are unable to use this printer's single graphics command ⟨ESC⟩ "*" for selecting the proper density. To maintain compatibility with this software, there are individual commands to select each of this printer's common graphics densities. These commands, which are shown in Table 7-4, can be used interchangeably with the corresponding ⟨ESC⟩ "*" command. Like the commands you are already familiar

with, these new commands are followed by two bytes to specify the number of graphics data bytes to print and then the data.

Table 7-4
Alternative graphics commands

Density	Single command	Individual command
8-dot noraml	⟨ESC⟩"*" CHR$(0) *n1 n2 m1 m2* ...	⟨ESC⟩"K" *n1 n2 m1 m2* ...
8-dot double	⟨ESC⟩"*" CHR$(1) *n1 n2 m1 m2* ...	⟨ESC⟩"L" *n1 n2 m1 m2* ...
8-dot double at double-speed	⟨ESC⟩"*" CHR$(2) *n1 n2 m1 m2* ...	⟨ESC⟩"Y" *n1 n2 m1 m2* ...
8-dot quadruple	⟨ESC⟩"*" CHR$(3) *n1 n2 m1 m2* ...	⟨ESC⟩"Z" *n1 n2 m1 m2* ...

REDEFINING ALTERNATE GRAPHICS CODES

At the end of this chapter, we'll discuss one mode that the printer offers to help you solve potential graphics problems. A redefining code allows you to change the density for graphics programs that use one of the four alternate codes. The command looks like this:

⟨ESC⟩ "?" *n0 n1*

Where *n0* is one of the four letters, "K", "L", "Y", or "Z", and *n1* is one of the numbers used with the ⟨ESC⟩ "*" command, 0 to 4, 6, 32, 33, 38, 39 and 40.

This is a quick way to change the aspect ratio of the design that you are printing. Changing the graphics mode will change the width without changing the height. However, you should make this change with caution.

If you change one of the 8-dot graphics mode to a 24-dot graphics mode, without changing the program that supplies the graphics data, you will print garbage (if the program prints at all). Remember, the 24-dot graphics modes require three times as much graphics data as the 8-dot graphics modes, and also, the data is arranged differently.

MEMO